Melete: Exploring the Components of Mixed-Initiative Artificial Intelligence Pipelines for Level Design

Sokol Murturi[1,2,*,†], Tony Pellicone[1], Matthew Yee-King[2] and Marco Gillies[2]

[1]Falmouth University, Faculty of Screen, Technology and Performance, Cornwall, United Kingdom
[2]Goldsmiths University of London, Department of Computing, London, United Kingdom

Abstract

Recent advancements in artificial intelligence(AI) and human-computer interaction (HCI) have led to the creation of innovative artifacts that bridge these fields, fostering creative collaboration between human authors and artificial intelligence. These advancements have found applications across a wide range of academic disciplines. Among these developments, many tools and frameworks have emerged to support the design and development of creative endeavors. However, there remains no clear consensus on how to structure these pipelines or what components should be included. To address this gap, we conducted a qualitative user study with twelve participants, examining how users engage with key elements of a mixed-initiative artificial intelligence (MIAI) pipeline for game development, including the procedural content generation (PCG) algorithm, its output, the user interface, playtesting, and the overall pipeline. Through an inductive thematic analysis, we developed a mixed-initiative interaction model, offering valuable insights into MIAI pipeline classification and guiding developers in designing more effective and user-centred pipelines.

Keywords

Creativity Support; Games/Play; HCI for Development; Qualitative study that explores concepts surrounding Mixed Initiative Artificial Intelligence

1. Introduction

The integration of AI into nearly every aspect of our daily lives has been driven by sophisticated algorithms and advanced statistical models. AI continues to play a pivotal role in shaping societal progress, particularly within the creative sector [1]. In academia, the growing interest in AI's role within creative industries has fuelled significant advancements—especially in the realm of video game research. Academic interests in video- games expands beyond computational research with other fields such as psychology [2], economics [3], and historiography [4], each exploring the impact of video games on our lives. Beyond their cultural significance, video games also serve as a testbed for innovations in machine learning[5], texture synthesis[6], PCG[7], image[8], narrative design[9], 3D asset creation[10], procedural animation[11], and audio production[12]. A growing area of academic research in games is the applications of PCG[13, 14].PCG enables on-the-fly creation of levels, characters, and textures. Games like Dwarf Fortress and Rogue leveraged these techniques to become landmark titles. Barton et al. note, "It was the first game to offer a character creation system based on a series of questions about moral dilemmas" [15]. Today, PCG has evolved significantly, employing neural networks and specialized algorithms to address the challenges of dynamic content generation. Recent research has focused on developing PCG systems aimed at delivering novel gameplay experiences. However, several persistent issues hinder the broader adoption of PCG. These include the technical expertise required to implement neural networks and PCG algorithms, and limiting accessibility for non-specialists.

Synergy 2025 HHAI-WS 2025: Workshops at the Fourth International Conference on Hybrid Human-Artificial Intelligence (HHAI), June 9–13, 2025, Pisa, Italy

sokol.murturi@falmouth.ac.uk (S. Murturi); Tony.Pellicone@falmouth.ac.uk (T. Pellicone); m.yee-king@gold.ac.uk (M. Yee-King); M.Gillies@gold.ac.uk (M. Gillies)
https://www.falmouth.ac.uk/staff/sokol-murturi (S. Murturi); https://www.falmouth.ac.uk/staff/dr-tony-pellicone (T. Pellicone); https://www.gold.ac.uk/computing/people/m-yee-king/ (M. Yee-King); https://www.gold.ac.uk/computing/people/m-gillies/ (M. Gillies)
0000-0001-9466-8981 (S. Murturi); 0000-0002-9774-2953 (T. Pellicone); 0000-0001-6606-2448 (M. Yee-King); 0000-0002-3100-9230 (M. Gillies)

Other challenges include non-transferable, project-specific algorithms [16], and the data-intensive nature of many AI-driven systems [17]. Accessibility remains a key barrier, for non-programmers, the construction of a constraint satisfaction PCG system—or any PCG framework—presents a formidable technical challenge [18]. Additionally, many PCG systems are difficult to repurpose. The content generation code written for one game is often not generalizable. Moreover, PCG systems based on neural networks require vast amounts of training data. Although techniques like adversarial neural networks show potential, they still rely heavily on substantial initial datasets [19]. Despite their promise, the barrier to entry for utilizing AI in creative processes remains high. Researchers have developed more user-friendly interfaces for PCG systems [20, 21], known as MIAI pipelines.

In this paper, we explore MIAI as a method for content generation that bridges the gap between programmers and non-programmers in the context of game design. MIAI pipelines combine graphical user interfaces with AI, allowing for complex, collaborative interactions between humans and machines. These systems can rapidly generate novel content while remaining accessible to users without technical backgrounds. However, more research is needed to understand how users engage with MIAI systems and how these interactions influence MIAI system design. This paper investigates the components that contribute to the effectiveness of a MIAI pipeline and the ways in which such systems can support creative workflows. Specifically, we examine how MIAI can assist designers in analyzing and iterating on video game content. To explore this, we use Melete[22]—a MIAI pipeline that helps designers to seed ideas, explore concepts, and refine implementations quickly with creative agency that merges the speed and novelty of PCG with the feedback and flexibility of a rapid prototyping environment. Melete enables the creation and testing of video game levels for asymmetrical 3D multiplayer environments, supporting both creativity and accessibility.

2. Previous Work

This background section explores two key areas of research: level design as a creative process, and the structure and components of existing MIAI pipelines. However, other key factors such as HCI and the types of PCG systems used in MIAI are explored in greater detail in our paper ”Melete: Playtesting and 3D Environments for Mixed-Initiative Artificial Intelligence as a Method for Prototyping Video Game Levels.”[22]. The foundation of any MIAI system relies on PCG to generate content so that a user may interact with it. There are many successful methods for implementing PCG [23, 24, 25]. These methods provide a cost and time-effective solution for generating content for games client side. Togelius et al. [13] presents an in-depth analysis of 14 separate academic studies. These studies used different forms of client-side PCG techniques to generate content for a multitude of video-game genres. Other examples of work like Evolutionary Dungeon Designer [26, 27, 28] are examples of ”online” generative AI. MIAI falls into the category of ”offline” content generation, where PCG is used during the development process.

2.0.1. Level Design

The work of a level designer is fundamental to the development process of most games because a level is the player's most direct experience with the mechanics of the game [29, 30, 31, 32, 33]. Levels in games are also where the aesthetic themes of a game and the narrative are best conveyed[32], which means that a skilled level designer is effectively tapping into all of the primary ways that games structure the player experience [34, 31]. However, level design is costly and complex [30, 33], which means that tools which aid the designer are always seen as appealing for game development teams of all types [35, 36]. Apart from improving production pipelines, PCG levels are also a mechanical feature of many games [36], with the creation of elegant modular level elements representing its own unique set of design considerations. This section will discuss level design as a discipline, review work related to PCG in level design, and then describe considerations for mixed initiative AI as a tool to aid with level design.

2.0.2. Level Design as a Practice

Level design comes from a tradition of valuing and elevating space in gameplay. Level design is often a means of both teaching and enhancing a player's understanding of the mechanical layer of the game, since it's a literalization of these mechanics in the form of a physical and intractable space. This sits alongside the purpose of a level in transmitting thematic information through the aesthetics of the space, and narrative information in terms of various forms of environmental storytelling [32]. This is to say that level design sits comfortably at an intersection between the elements of a game that comprise player experience [31].

The focus of a level designer in a game development pipeline tends to be on player behavior. A typical workflow for level design in a commercial project will generally follow several steps: planning in pre-production, design of related systems that will play into the functional requirements of the level, higher abstractions of layout, blockouts that give a playable rough draft, scripting where core functionality is added (e.g. buttons, triggers, doors), lighting, environmental art, and finally release [33]. The level designer serves as the front line in terms of communicating and mediating the essential gameplay to the player, and thus must create spaces that exemplify the player experience [30].

Level designers must also work both quickly and iteratively, as playtesting is a core aspect of effective level design [30, 32, 33]. Totten (2016) phrases this as such, "The goal of a level designer should be to get his or her design in interactive form as soon as possible. This way, he and others can playtest the level design, meaning that he plays the game to evaluate whether it fulfills its original design goals," (pp. 65 - 66). As the member of a team who is most directly responsible for trying the mechanics, aesthetics, and narratives of the games together in order to give the player meaningful gameplay actions [29], this also means the tools available to them must be both powerful and easy to use [33]. PCG, and recently MIAI have presented a compelling use case for augmenting the tools available to level designers to help assist both the technical and creative aspects of this practice [36, 37, 38].

2.0.3. Applications and Design Considerations of Mixed Initiative Procedural Generation to Level Design

PCG has a long history in game design as a set of tools that are used to give depth and flexibility to design. For example, the SpeedTree[39]software has been in use since 2003 to create realistic tree assets for games through procedural rules [40] L-systems. This has continued and been expanded over the past several years with a growing body of MIAI work related to game design, and level design in particular. Examples of this include Sentient sketchbook[41], Ropossum[20], and Tanagra[42]. which we shall explore later in this section. The previous work cited above presents several suggestions for the creation of mixed initiative software for level design. Designers will be coming into interactions with mixed initiative systems with prior expectations about what the AI side of the tool will be doing, ranging from a friend who is there to provide a fun and engaging brainstorming space, to a manager who is giving specific tasks and directives. Other scholars suggest that there are a number of roles that people tend to use to frame AI, and clear explications of what the AI is contributing to the work at hand are necessary to help designers in mixed initiative systems to guide their use of the tool [43, 44]. Defining user expectations for the AI side of a mixed initiative system is important because the user's requirements of a level editor will also change depending on their role in the development process (e.g. a designer on a development team versus a hobbyist/modder working independently) and the technical requirements of the level editor laid out by the mechanics of the game [30, 45, 33]. Aliaga et al.'s (2024)[46] work on mixed initiative modules for level editing in Unity suggest that designers are often open to these types of workflows, provided they are easy to use and fit in neatly with the mechanical requirements of the game content that they are developing. Previous work on the creation of cities in a sim game (which is distinct from level editing, but shares key characteristics in terms of creating a gameplay space based on mechanical constraints) indicates that there is not a clear impact on creativity either way, but that the technical affordances of speeding up work are seen as a clear benefit to this type of system [38, 22]. A key consideration in terms of balancing and easing creation versus creative control is the ability of the designer to both understand and edit the algorithm, tuning it towards what makes the most sense for the mechanics that they are trying to convey [46, 47, 48, 22]. While editability and explainability are both key components of a mixed initiative system, the need of a user to constantly mind the AI partner can lead to fatigue. This comes from the cognitive load of making a series of taxing choices, and can be designed around by pruning selections based on previous choices the user has made, crowdsourcing of choices, and by speeding up the process (reducing human wait times) through limiting potential choices [47]. Altogether this indicates that mixed initiative tools for level design have a great deal of promise, but that expectations need to be clearly set, there needs to be opportunities for the designer to tune the output of the systems to respond to mechanical constraints of the desired player experience, and there needs to be various ways to mediate the options available to the designer to prevent cognitive fatigue over multiple iterations. Melete enables designers to perform these tasks, but the exact mechanisms remain somewhat unclear. This uncertainty highlights the need to examine the various components of MIAI systems to determine which elements are effective and which are not.

2.0.4. Mixed-Initiative Artificial Intelligence Pipelines and their challenges

As an emerging research area, MIAI focuses on improving HCI with AI[49, 50, 51]. The concept was first introduced by Carbonell in the 1970s[52], with early implementations like Dawkins' Biomorphs[53] and Todd & Latham's Mutator[54], which incorporated UI elements for evolutionary exploration of genotypes. Recent efforts have worked to formalize and define the concept of MIAI [55]. However, it is generally accepted in games research that MIAI pipelines bridge the gap between designers and programmers by offering intuitive interfaces for applying gameplay semantics and constraints to small datasets, enabling iterative content generation and refinement. While some usability studies exist, they remain limited due to small sample sizes and the time-intensive nature of experimentation. Liapis et al.[56] provide a detailed overview of MIAI in game design, while Amershi et al.[57] discuss its broader application. Lai et al. [58] examine its industrial relevance. MIAI emphasizes AI as a collaborative tool rather than a sole creator, aiming to blend human creativity with AI's computational capabilities to balance innovation with production efficiency. The multifaceted nature of MIAI pipelines—encompassing PCG, HCI[49, 50, 51], and game design—poses significant challenges for their evaluation and advancement. Each of these components demands distinct analytical approaches. For instance, PCG research often emphasizes output evaluation, as outlined by Togelius et al. [13], while HCI prioritizes user interaction and usability. Meanwhile, level design is typically examined through game user research [59](GUR) methodologies. As a result, disentangling the contribution of each component within a unified MIAI system proves difficult in a single study. In this paper, we address this complexity by examining the individual roles of PCG, HCI, and level design within MIAI pipelines. We propose methods for isolating and evaluating these components using established techniques from their respective domains, thereby providing practical guidance for MIAI developers seeking to test and improve specific elements of their systems.

2.0.5. Mixed-Initiative Artificial Intelligence Pipelines and their structures

In this section we will present some notable MIAI works and place their components. The Sentient Sketchbook [60, 61] is an MIAI tool that assists designers in creating low-resolution maps for real-time strategy (RTS) games. It leverages AI to analyze map topology and uses A* pathfinding to visually and statistically inform users about resource distribution and gameplay balance. The components of Sentient sketchbook are an interaction loop for editing and selecting PCG output. Another notable MIAI tool is Ropossum [62], developed for designing physics-based puzzle games like Cut the Rope. Ropossum stands out for tackling the complexities of physics-based gameplay. It combines an evolutionary content generation framework with a playability constraint solver to ensure all generated levels are solvable while continuously refining the algorithm. The third example, Tanagra [42], supports platformer level design through a mix of reactive planning and numerical constraint solving. It autonomously generates levels but also adapts in real-time to designer input, maintaining both creativity and playability. Tanagra uses the Choco constraint solver to control geometric layout, demonstrating how MIAI can streamline design while ensuring high-quality content.

Lode Enhancer[63] is an AI-assisted level design tool for creating Lode Runner levels through intelligent upscaling. Designers draw on one of three canvases (4x4, 8x8, or 16x16), and changes are synced across all scales via a scaler module. A tile toolbar aids design, while a persistence slider controls which tiles remain fixed during upscaling.

In Procedural Level Generation in Educational Games From Natural Language Instruction[64], the authors present an end-to-end framework that transforms natural language directives into playable game levels through four key components. The NL Instruction Parser extracts essential details like theme, metrics, and complexity. The Game Level Generator uses this input to create level candidates aligned with educational goals. The DRL Level Evaluator assesses difficulty using deep reinforcement learning. Finally, the Playable Level Generator converts the selected level into a 3D environment within the FUTURE WORLDS Unity platform.

Germinate[65] is a mixed-initiative game design tool for creating rhetorical games, built on the Gemini generator. It features a browser-based interface where users specify desired game properties, which are translated into structured constraints (Gemini intents). The system then generates a batch of playable games based on these inputs. Users can explore, play, and refine the games through an iterative loop of adjusting constraints and regenerating content. Germinate simplifies the process by offering structured intent editing and live feedback, reducing common errors and improving usability.

Morai Maker[66] is a Unity3D-based level design tool inspired by Mario Maker, designed to explore AI-assisted creation in the Super Mario Bros. domain due to its PCGML relevance and broad familiarity. The interface features a central level view, minimap, sprite palette, and an ”End Turn” button that lets users alternate turns with an AI partner. The AI suggests additions, shown step-by-step, with the camera following changes. Users can playtest the level at any time, and all actions are logged to track human-AI contributions.

While all these systems are classified as MIAI pipelines for game development, they differ significantly in structure and functionality. For instance, Sentient Sketchbook uniquely provides balance feedback, while Tanagra can fully override designer input. These differences make direct comparisons challenging and highlight that further study into MIAI pipelines is important. Particularly understanding the role of each component of an MIAI pipeline and its possible testing methods.

2.1. Melete

2.1.1. The structure of Melete

In the paper: “Melete: Playtesting and 3D Environments for Mixed-Initiative Artificial Intelligence as a Method for Prototyping Video Game Levels”[22] we expand further on the implementation of Melete. An overview of the system is presented below. In figure 1, we can see the two distinct phases of Melete: the Kernel creation phase and the interaction loop. During the Kernel creation phase illustrated in figure2, designers work with the WFC algorithm to create a Kernel. This Kernel allows the WFC algorithm to generate artefacts in a style similar to the designer’s. The WFC algorithm implementation is inspired by the computer graphics technique known as texture synthesis, which traditionally repairs and corrects digital images. However, what differentiates WFC from other texture synthesis techniques is that it does not merge or average adjacent pixels. This feature enables the use of a palette to preserve gameplay semantics.

To interact with the wavefunction collapse (WFC) algorithm, designers are provided with a palette of game objects. Using this palette, they can create a small-scale 3D environment within a 10×10 Kernel input grid. This input grid helps the WFC algorithm develop a table of possible tile combinations—specifically, 2×2 sets of game objects that appear within the Kernel input. Additionally, the algorithm generates a likelihood score for the placement of certain tiles and builds a dictionary of possible overlapping tiles. While Melete is primarily tailored for generating 3D environments, abstracting game objects

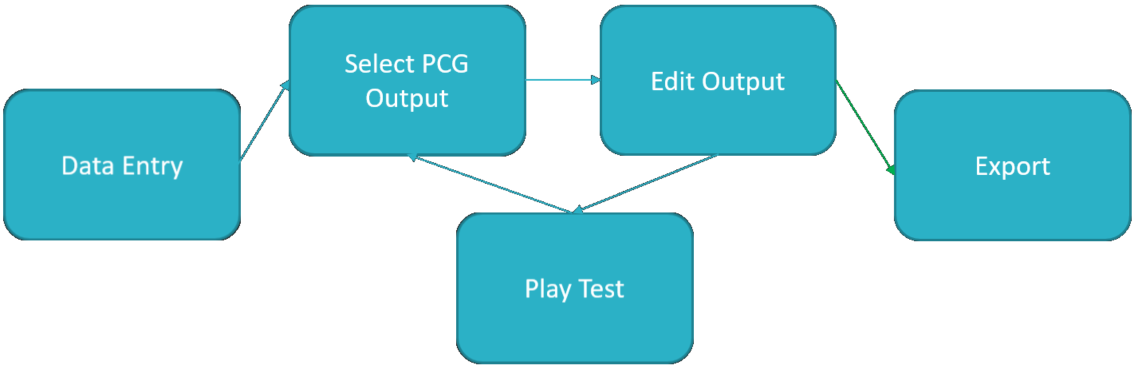

Figure 1: Melete Overview

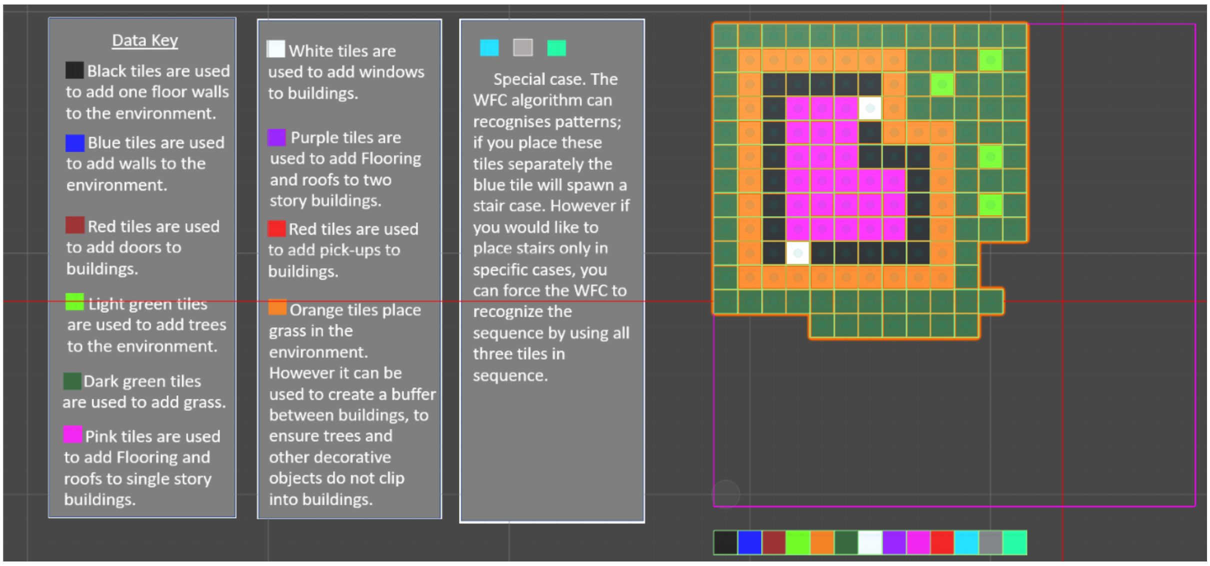

Figure 2: WFC Kernel Interaction

from the PCG algorithm means the palette can represent various gameplay elements or environmental components. This abstraction allows designers to focus on crafting the environmental topology, while the PCG algorithm manages the broader structure of the 3D environment. Once designers finalize the training input for the WFC algorithm, they enter the interaction loop illustrated in figure3. Upon entering the interaction loop, users begin by selecting an output generated by the WFC algorithm. They can cycle through multiple outputs until they find one that appears interesting. Since PCG outputs do not always generate immediately usable content, users can move into the next stage: modifying the WFC output. As they make changes, users can explore a 3D representation of their environment, which uses the palette's game objects to place environmental semantics. An avatar is also provided, enabling users to explore the generated environment firsthand. Throughout the interaction loop, users can freely interact with any of these three components as they see fit. Once satisfied, the completed environment can be stored in XML or JPEG format for export.

2.1.2. Melete: Playtesting and 3D Environments for Mixed-Initiative Artificial Intelligence as a Method for Prototyping Video Game Levels

While exploring the existing literature on MIAI, we noticed that many systems developed to date did not allow or emphasize exploration of the generated content. One of the key components of developing

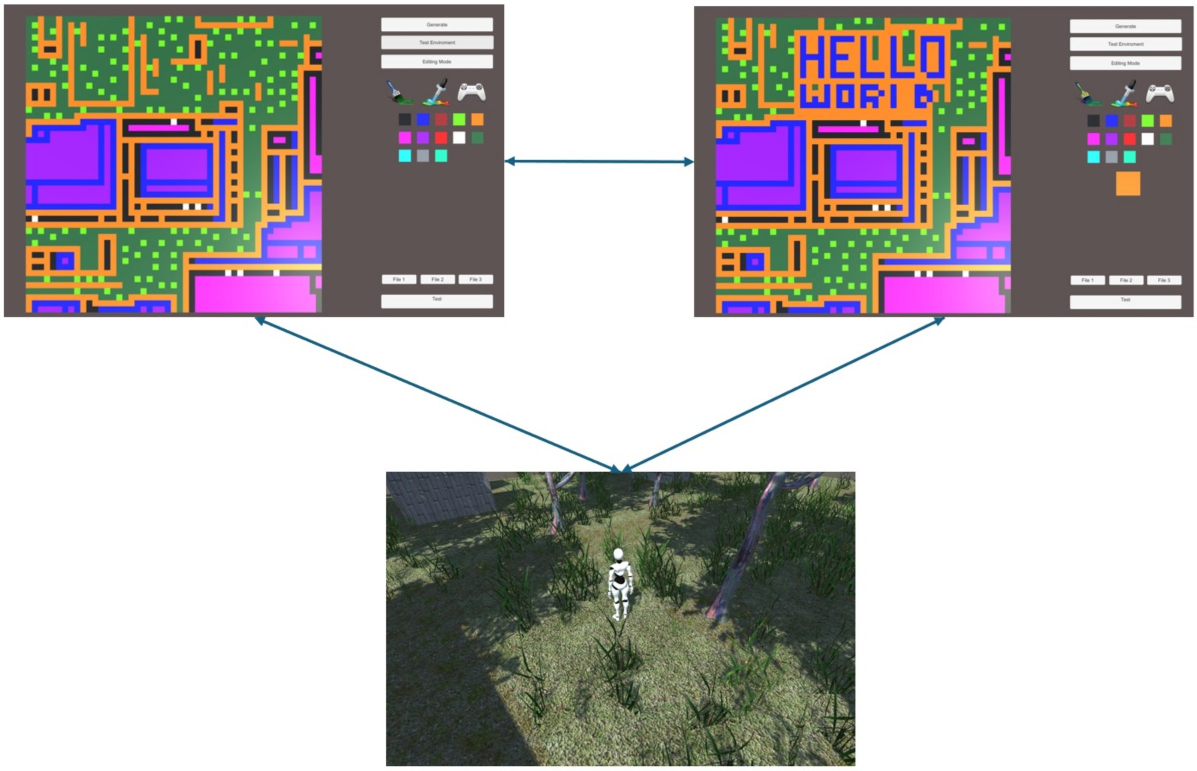

Figure 3: Interaction Loop

virtual environments is iterative design. To support this, Melete introduces a novel MIAI pipeline that gives users the opportunity to explore the content they generate through an avatar. Along with validating Melete as a useful MIAI tool, we sought to highlight the importance of playtesting in MIAI systems. Across several papers cited in the above MIAI component section, a key validation method across these studies was conducting expert analysis. In our study, "Melete: Playtesting and 3D Environments for Mixed-Initiative Artificial Intelligence as a Method for Prototyping Video Game Levels,"[22] we conducted an expert analysis involving five experts from industry and academia specializing in artificial intelligence and game design. The analysis underscored the critical role of playtesting in the design process, emphasizing the need for genre-specific playtesting components within MIAI pipelines. It also affirmed that Melete is a useful tool for prototyping video game environments. However, it was noted that the role of MIAI pipelines should remain focused on prototyping creative outputs rather than extending into full production.

Method To evaluate the components of the MIAI interfaces and the overall system, we designed five testing conditions. The first three focused on individual elements of the MIAI system: the interface, the procedural content generation (PCG), and data entry. The fourth and fifth conditions assessed the implementation of the full MIAI interface, specifically examining the interaction loop and the complete Melete system, including the data visualization input system. A base dataset, featuring common elements from the Battle Royale genre such as multi-level buildings, ruins, and forests, was created for these tests. This dataset was used consistently across conditions involving interaction with the wave function collapse algorithm, with users free to modify it as needed. These conditions are illustrated in table1.

In the first condition, the focus was on evaluating the Melete interface independently of its generation capabilities. Participants were tasked with designing a map from scratch using only the interface, without any input from the procedural generation component. This setup allowed for a clear assessment of the user interface within the MIAI pipeline and was particularly suited for UI/UX investigations,

	Kernal	PCG	Edit	Play
Condition 1			X	X
Condition 2		X		X
Condition 3	X			X
Condition 4		X	X	X
Condition 5	X	X	X	X

Table 1
MIAI pipeline condition table

helping MIAI pipeline developers better understand user interaction and usability. In the second condition, participants were asked to choose a map generated by the wave function collapse algorithm that best aligned with a given design brief. This condition involved minimal user input, emphasizing the generative capabilities of the system from a human-centered perspective. It offered valuable insights into how users perceive and evaluate algorithmically generated content, and could inform the development of generation methods that require low cognitive effort from users. In the third condition, participants engaged directly with the wave function collapse algorithm's kernel, as presented in our Melete outline, this is used to influence the generation process and produce an output they believed would meet the design brief. This condition emphasized user-driven refinement of generative content and was particularly useful for identifying missing design elements within a game. It would be well-suited for GUR studies focused on designers expectations and creative control. In the fourth condition, participants engaged with the interaction loop outlined in the Melete system, allowing them to generate, edit, and test maps. However, they were not permitted to modify the wave function collapse algorithm's dataset. In contrast, the fifth condition granted participants full access to all aspects of Melete, including dataset adjustments. These conditions highlighted the dynamics of human-computer interaction within AI-supported design and are particularly useful for comparing different MIAI pipeline models to explore how various system configurations influence user experience and design outcomes. Although we wanted participants to compare their experiences across different conditions, we chose to limit the number of conditions each participant engaged with, while extending the duration of each session. Unlike other MIAI studies in the literature, which often impose strict time limits for interaction with the pipeline, our focus was on the participants' experience and how they evaluated individual components in relation to the full system. This approach aimed to gather richer qualitative feedback. Each participant completed two conditions: one randomly assigned from conditions one to four, and condition five as their second. For each condition, participants had 15 minutes to complete their task, with an optional 5-minute extension for final adjustments. After each session, they completed a short survey and participated in a speak aloud interview to reflect on their experience.

2.1.3. Participants

To recruit participants with relevant experience in computing and game design, a two-pronged approach was used. First, students were invited to participate during computing courses, with a short demo of Melete provided to spark interest. Second, the principal researcher approached students during computing and game development classes. This method ensured a diverse and capable participant pool without bias. Twelve participants took part in the study—six male and six female—providing a balanced and diverse sample. Eleven were aged 18–25, and one was between 25–35. Nine were undergraduate students, while three were postgraduates, all studying in computing-related fields, such as business computing and virtual/augmented reality. All participants had gaming experience, with most regularly playing and being familiar with the Battle Royale genre. This group enabled the collection of rich qualitative data.

2.1.4. Materials

During the experiment, the principal researcher supplied all necessary materials. This included a copy of Melete running on the latest version of Unity, as well as input devices—a mouse and an Xbox joystick—to facilitate user interaction with the system.

2.1.5. Procedure

Each participant was brought to the principal researcher's office and provided with a research handout. After reading the handout, participants were given the option to withdraw or continue, with the principal researcher available to answer any questions before the study began. Participants were then randomly assigned an initial condition and given 15 minutes to complete the task, with the option to extend their time—a choice many took advantage of. Following the first condition, participants completed a short survey and took part in a semi-structured interview. They then moved on to the second condition, which was either Melete (if not used in the first round) or another randomly assigned condition from the remaining options. The same task was attempted under similar time conditions, with extensions allowed if needed. Afterward, participants again completed a survey and interview focused on their experience. Throughout the experiment, the principal researcher was present to troubleshoot any technical issues and provided non-technical support during the debrief. In the debrief session, the researcher explained the purpose of the study and answered any remaining questions.

2.1.6. Qualitative data gathering

To collect qualitative data, a semi-structured, think-aloud interview was conducted by the principal researcher with each participant after they had completed both the task and the questionnaire for a given condition. During the interview, participants reviewed a recording of their interaction with Melete and explained their actions, while also responding to a set of open-ended questions designed to guide discussion:

- How could information about the level design process be presented better by this system?
- What would make you want to use this system in an actual design process?
- What made you not want to use this system in the design process?
- What improvements could be made to the system so you could better understand how to use it?
- If you felt uncomfortable during the task, please indicate the reasons.

Additionally, if the participant had completed condition 5, they were also asked:

- How do you feel your experience in the two conditions differed?

These questions, inspired by the Microsoft Single Ease of Use Questionnaire (SEQ), were not intended to elicit specific answers, but rather to facilitate open-ended discussion and deeper insight into the participant's experience with the system.

3. Results

Given the richness of the qualitative data collected, we chose to analyze our findings using thematic analysis as outlined by Clarke et al.[67]. Following the interviews, we began by familiarizing ourselves with the data—re-watching participant interviews, transcribing the discussions, and taking notes to highlight key feedback. During this initial phase, the focus was not on identifying themes, but on pinpointing significant quotes for deeper analysis. In the second phase, we revisited the transcripts to generate codes and began grouping these codes to identify overarching themes. This process led to the definition of three key themes that influence users' ability to interact with the MIAI system: Control, Design and play, as shown in figure 4. Each theme was shaped by several underlying factors, captured through the codes, which helped determine how each theme impacted user interaction. The codes themselves were presented in a neutral manner. Although we also conducted a quantitative analysis, this paper focuses on the presentation of the qualitative findings.

The Quantitative data has been added to the appendix, the questions that where used to conduct this quantitative analysis where based off of the Microsoft Single Ease of Use Questionnaire. This is a brief summary of the quantitative data which can also be found in the appendix. As participants only took part in the Melete condition and another randomly assigned condition, the data is labeled as Melete responses and Other condition responses. When comparing the Melete responses and the other condition responses, several trends emerge. Overall, both datasets reflect a positive experience with any version of the system, but the Other Condition responses dataset reveals slightly more variability and lower average scores in several areas. In terms of enjoyment, the mean dropped slightly from 4.42 to 4.25, and the standard deviation increased from 0.67 to 0.97, indicating more varied responses in the other conditions group. The success rate also declined, with the mean falling from 4.09 to 3.82 and a modest increase in variability. Similarly, participants in the other conditions dataset reported slightly less ease in using the system and found it marginally less helpful in the design process, as shown by decreases in average scores for both "usage extent(Q10)" (from 4.08 to 3.92) and "helpfulness(Q8)" (from 4.25 to 3.92). The increase in standard deviations across all categories in the other conditions dataset suggests a broader range of user experiences, when using different combinations of the Melete tool potentially pointing to inconsistencies in how the other system performed across different participants.

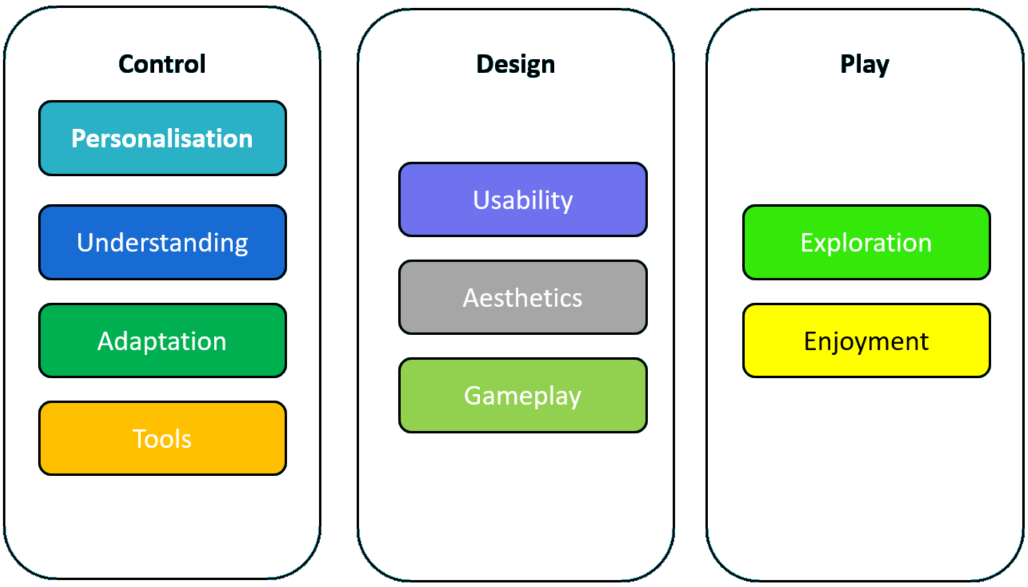

Figure 4: Codes visualisation

4. Thematic analysis

One of the key challenges in presenting this thematic analysis is the complexity and richness of the qualitative data, which makes it difficult to fully disentangle and concisely present all themes. As a result, some themes may appear interwoven in this summary. However, a full transcript of the thematic analysis is available for a more detailed exploration of each theme and its corresponding codes. Here we will highlight some of the more interesting themes and their codes.

4.1. Control

The theme *Control* explores the user's ability to effectively work with the system to complete the design task. It incorporates the codes: *Personalization/Relating to past experience, understanding, adaptation,* *and tools*, examining how participants engaged with and manipulated various components of the MIAI system across both the initial conditions and when using Melete.

4.1.1. Understanding

This code identifies factors affecting users' comprehension of information presented by the system. For example, Participant 3 (Condition 1) demonstrated partial understanding but noted UI clarity issues: *"...I looked at my thing and said, okay, I just want a wall here. Then I realized later on I had to paint everything black..."* In addition, Participant 11 (Condition 3) expressed confusion: *"It was not clear enough... it's more like technical information and don't think it's like very user-centered."* Both responses highlight the importance of clear data representation in MIAI systems.

4.1.2. Personalization

The code *Personalization/Relating to past experience* refers to the user's ability to draw from personal or gameplay experience when designing with MIAI systems. Many participants leveraged their understanding of game design and mechanics to inform their decisions. *Participant 1, Condition 1:* "I was just thinking back to when I played Fortnite... I know how people tend to flock towards the biggest buildings... So I thought about how people played and I made that (the map)..." In Condition 5, the same participant reflected further:*Participant 1:* "...But maybe for a game like Halo or Doom this is kind of okay but for a battle royale with open space... from my experiences, like a really big part of it..." *Participant 1:* "Yeah. I feel like if my brother, he doesn't camp, if he had been using this he would have seen that building and thought we'd start again... I saw a big building. That's perfect... lots of cover." These responses illustrate how the participant applied personal gaming experience to their design choices. They also demonstrate crossover with other codes, including *aesthetics, adaptation*, and *exploration*, showing a nuanced understanding of how the tool could support personalized and strategic map creation.

4.1.3. Adaptation

The code *adaptation* refers to how participants adjusted their design approach in response to new information discovered through system exploration or play-testing within the interaction loop. As previously noted, users often evolved their strategies between conditions. A strong example comes from Participant 8 in Condition 5. Initially struggling with the output, they realized that the generated building lacked entrances. After identifying this issue during play-testing, they returned to design mode to fix it: *Participant 8:* "...with the ones we put blocks and space that I put together, it made a massive, um, building. But then I couldn't get out of the building because there was no door... Then I placed in the doors... I was able to free flow like in and out with that..." This adaptability illustrates how users refined their designs in response to system feedback. It's also important to note that missing functionality—such as essential tools—can limit the user's ability to adapt effectively and impacts overall control within Melete.

4.1.4. Tools/Functionality

Originally categorized under *Design*, the *tools/functionality* code was reassigned to the *Control* theme, as it more directly impacts users' ability to effectively interact with and manage the design process. This factor also emerged during play-testing, where users encountered limitations due to missing or unclear features. An illustrative example comes from Participant 3 in Condition 1:

> "Um, yes, for the most part. I mean I did lose myself when I was going in and out and not realizing where I spawn and what's the consent... when I added the stairs and went back in and not realizing from this camera perspective not knowing exactly where I was... ...Yeah, I mean a mini map might be helpful and relatively straightforward to put in?... the other

> thing would be when you're going to test-environment, if I could choose where I spawned... that would be a big deal cause it's, I need to transition."

This feedback highlights the importance of seemingly small tools—such as a mini-map or spawn point selection—that can significantly improve user orientation and control within the system. The absence of these features impacted the participant's ability to navigate and refine their design efficiently.

4.2. Design

The theme *Design* focuses on the overall aesthetic, gameplay, and usability aspects of Melete. A key code within this theme is *gameplay*, which addresses both implemented and missing mechanics, and how these affect the user's ability to complete the task. While it would have been possible to separate these into distinct codes, we chose to explore them together, as users often evaluated design limitations holistically—adapting their design approach in response to the system's constraints or affordances. Participant 8 (Condition 4) highlighted how existing elements inspired their design:*Participant 8: "I think it's necessary... if it was just a blank page, I wouldn't have the same inspiration... it gives me an idea of, okay, so this is roughly how I was using the tools."* Gameplay mechanics were also noted by Participant 9, who focused on avatar movement within the testing environment: *Participant 9:"...Oh by the way, it's really cool how the thing moves... it goes like that." Participant 9: "So you know how the person moves... I got so fascinated with it I kept doing it... how it moves... your camera takes the turn in a curve."*These reflections show how visual feedback, mechanics, and movement design all contribute to the usability and creative inspiration within Melete.

4.3. Play

The final theme, *Play*, explores whether users enjoyed using the system and how actively they engaged with its tools during the design process.

4.3.1. Exploration

Exploration refers to a participant's willingness to experiment with different features and interactions. Participant 2, in Condition 5, demonstrated this through their creative use of system tools: *Participant 2: "Yeah, I think it was the, um, the interactivity... So after you get from here where there were rocks and stuff, you just have to run directly at the building... with a second I could interact and make this little corridor after the cover so you could find your way..."*

4.3.2. Enjoyment

Enjoyment considers whether users found the experience positive or negative. Participant 2, also in the Melete condition, expressed enthusiasm about the system's generative potential: *Participant 2:"I feel like the generation would help... because it could be to a point where you went to generate different types of terrain... once you've found like, Oh, I want it to be a castle, you'd find a castle and then you could add onto it and make it how you wanted it to be."*The participant's tone, as heard in the audio, conveyed clear excitement and engagement, reinforcing the positive emotional impact of using Melete for creative design.

5. Discussion and Mixed Initiative Interaction Model

Following our thematic analysis, we developed a model to describe participant interactions with MIAI pipelines, shown in Figure 5.

In the previous section, we presented all of the themes identified in this study and provided examples of qualitative responses that helped reveal those themes. These themes have been grouped into the following categories:

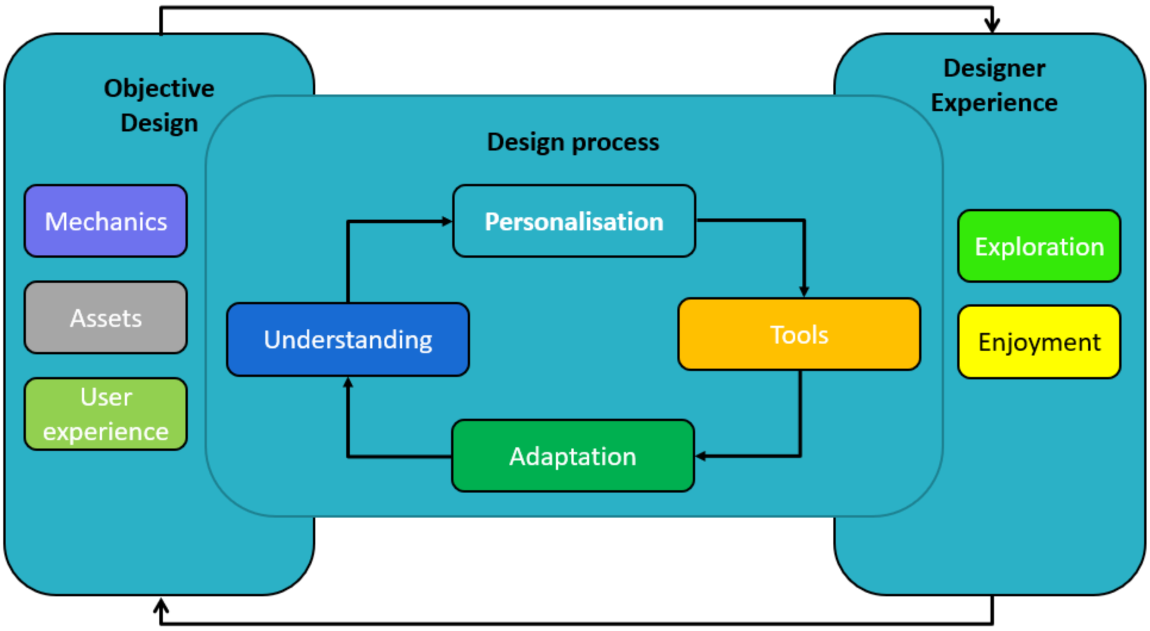

Figure 5: Mixed Initiative Interaction Model

- *Objective Design* – Themes relating to what the AI/system/pipeline is intended to affect.
- *Design Process* – Themes over which the user has control.
- *Designer Experience* – Themes concerning how the agent and human interact.

These categories are linked to specific methodological approaches for improving AI implementation, the system's underlying rationale, and HCI interaction.

The purpose of this model is to help researchers identify key areas of interest in MIAI pipelines by breaking down qualitative data into identifiable themes. It is important to note that participants in qualitative studies may produce both positive and negative comments about a given theme. As such, the model not only identifies areas for improvement but also highlights where MIAI pipelines are performing well.

To apply this model, researchers may conduct small-scale qualitative studies with MIAI pipelines. We recommend following the methodological procedure outlined in Sub-section 2.1.6. After conducting a study and coding the thematic analysis, researchers can use the model to highlight areas of consideration: Design Process (Objective-specific tasks), Objective Design (System-specific tasks), and Designer Experience (User experience).

- **Objective-specific tasks** – These refer to what the individual designing the system is trying to automate. In our case, Melete aims to automate the process of populating a small 3D environment.
- **System-specific tasks** – These refer to the goal of the person using the system. In our case, users were trying to develop a level for a Battle Royale-style game.
- **User experience** – This refers to how easily the user can interact with the system.

Before developing any MIAI pipeline, researchers should consider the following questions:

- In relation to objective-specific tasks: What specific functions or processes is the AI responsible for within the system?
- In relation to system-specific tasks: What is the end user intended to produce or accomplish using the system?

- In relation to user experience: How does the user engage with both the system and the AI, and what is the nature of that interaction?

With regard to how the Mixed Initiative Interaction Model can support system improvement, the following list outlines methodological approaches based on receiving negative feedback within specific categories:

- Issues in **Objective Design** suggest using AI testing methods to verify that outputs meet their intended purpose.
- Challenges in the **Design Process** point to the use of GUR methods for system refinement.
- Problems with **Designer Experience** call for HCI methods to assess usability and user satisfaction.

While our focus is on the application of this model to MIAI pipelines in game design, the same framework can be extended to other research domains.

In the following sub sections we will use the Mixed Initiative Interaction Model to help highlight areas of improvement for conditions one through four.

5.1. Condition 1 - Level Editor

In Condition 1, participants were asked to design a map entirely from scratch—functioning much like a traditional level editor. This approach proved time-consuming and pushed users out of the iterative *Design Process* theme of the MIAI model, encouraging a more linear, intuition-driven method. As a result, participants often hesitated to fully engage with the available tools, leading to difficulties in approaching the task effectively.

For example: **Participant 6:** *"...I always like to finish [the level]... I would prefer to test it when it's complete, rather than at smaller stages..."* **Participant 3:** *"...This is more of a case of 'I've built levels in various level editors as well.' So creating something from nothing is the status quo..."*

Both participants initially made avoidable, time-consuming mistakes by relying on linear design logic. However, as they began interacting with the interface—particularly through play-testing—their understanding of the system improved. This highlights the importance of tactile feedback and iterative loops in supporting effective design.

While Condition 1 functions as a standard level editor, its true value in an MIAI context lies in offering real-time feedback, which can guide and refine user decisions within the broader interaction loop. This observation is also supported by our expert analysis [22].

In our interaction model, both of these comments fall under the *Design Process* category—specifically, problems related to understanding and adaptation. Although neither comment is negative it suggests that for the level editor condition, we could apply GUR methods to improve the how Meletes level editor worked.

5.2. Condition 2 - PCG Output

Condition 2 focused on using a PCG system to generate content, but its lack of adaptability and control pushed participants out of the *Design Process* theme and into a more linear workflow. **Participant 10:***"Yer, I think especially as an AI tool I could see myself using this system to help me in the design process..." "...But I would want the ability to draw in my maze[level] every time... it would be a good way to move this sort of tool forward."* Although participants recognized the usefulness of generation, the inability to modify outputs limited engagement. They relied on experience to choose suitable levels but lacked the tools to iterate.**Interviewer:***"Do you think that you were able to complete the challenge?"***Participant 10:***"I guess I did."* This uncertainty highlights the limitations of PCG systems as stand-alone tools—undermining user confidence and reducing the effectiveness of the design process. With respect to the interaction model, these comments fall under the *Design Process* category. However, they are largely positive reflections, accompanied by constructive suggestions—many of which were later implemented in the Melete condition. As the feedback is both affirmative and already addressed in a subsequent condition, no further intervention is necessary.

5.3. Condition 3 - Kernel interaction Only

In Condition 3, participants generated a map using a dataset of their own creation, offering partial control over the output. However, like Condition 2, it limited adaptability and pushed users into a linear design process, moving them away from the *Design Process* theme. Two main challenges emerged: the lack of editing tools hindered *Objective Design*, and the complexity of dataset creation made the system difficult to understand. **Participant 4:** *"...I actually, I think I read all of them. I thought these were more advanced, so I avoided them for now..."* This condition highlighted that, without clear explanations or accessible tools, users felt overwhelmed and were discouraged from engaging with the system. In terms of the interaction model, this feedback was coded as an issue with enjoyment. As it reflects a negative interaction with the system, it falls under the *User Experience* category. To address this, HCI evaluation methods should be applied. In future experiments, we plan to incorporate an onboarding tutorial to help participants better understand how the WFC algorithm works and how to manipulate its output. This is an example of a "Design Interaction Method" aimed at helping users build a mental model of how a system operates, as discussed in [68, 69].

5.4. Condition 4 - MIAI without Kernel

In Condition 4, participants engaged with Melete's interaction loop, encouraging an iterative design approach. This setup not only demonstrated how mixed-initiative systems support design but also made the process more enjoyable and exploratory. **Participant 8:** *"HAHA! That was the box like that... And then I realized I don't want walls... So I went back, check paint and then came back again."* The participant's laughter reflects a playful and engaging experience. Real-time feedback and the ability to iterate between design and playtesting reduced stress and encouraged experimentation, helping users visualize the end-user experience more effectively.When asked about procedural generation, the participant added: *"...if it was just a blank page, I wouldn't have the same inspiration... So that helped my understanding."* This highlights both the value and a limitation of the system: while generation inspired creativity, the lack of transparency in how content was produced made it difficult for users to focus on *Objective Design*, particularly in relation to mechanics, assets, and player experience. In terms of the interaction model, there are two areas that were commented on. The initial comments relate to the *Designer Experience* and are generally positive; as such, the interaction loop may be sufficient as an initial implementation, and no further HCI evaluation might be needed. In contrast, other feedback suggests that the WFC algorithm may not be well-suited to scenarios where there is no need to interact with the initial seed/kernel generation. It may be more appropriate to explore alternative PCG methods for this type of system.

5.5. Condition 5 compared to other conditions

In Condition 5, participants worked with the full Melete system, which showcased how a mixed-initiative interaction loop supports iterative design and provides greater insight into the generation process. This section compares participants' experiences with Melete against their previously tested conditions.**Participant 1** (previously Condition 1): *"If you're making like multiplayer maps... this is definitely very helpful rather than have to start from scratch every single time."* They emphasized Melete's efficiency over manual design. **Participant 2** (previously Condition 2): *"I'd use it for... an online D&D experience... The dungeon master would create the world and players could play through it."* They saw Melete's potential in creative, collaborative settings. **Participant 7** (previously Condition 3): *"It's never going to give me exactly what I want... Whereas with solution two, you can actually just clean up the rough edges yourself... it's very useful."* This participant appreciated how Melete balanced procedural generation with manual refinement, making it more production-ready. **Participant 8** (previously Condition 4): *"...I felt like I had more control... It just kind of allowed more freedom for my creativity with... all the other tools."* They highlighted the creative freedom enabled by combining data input control with design tools.Overall, Condition 5 offered participants more flexibility, efficiency, and creative control compared to earlier conditions, reinforcing the value of an integrated, iterative MIAI workflow.

6. Conclusion and future work

This study supports previous findings from [22] that Melete is a promising tool for prototyping levels in Battle Royale-style games. Future research could explore quantitative methods for evaluating MIAI pipelines and examine how kernel interaction impacts the design process—particularly the designer's sense of control. In addition to this, in the MIAI Pipelines section, we highlighted notable examples of MIAI systems in games. From this, we developed Table 2, which outlines common components found in these pipelines. These include methods for interacting with the kernel/source data used to train PCG algorithms, selecting and editing PCG output, testing interactions between the designer and AI, presenting feedback (e.g., balance data), and mechanisms for the AI to correct or override the designer. The table breaks down each system's components based on descriptions from their respective articles. When viewed alongside the MIAI interaction model, it reveals potential areas for future research. One such direction is comparing different MIAI systems. For instance, Melete and Germinate share many components but differ in implementation. Notably, Germinate uses natural language to generate kernels. Future work could explore comparing Melete with Germinate or with a version of Melete that incorporates LLMs for kernel interaction in the WFC algorithm. While participants were able to

	Kernel	PCG selection	Editing	Play	information	correction	Automated
Melete	X	X	X	X	0	0	0
Sentient Sketchbook	0	X	X	0	X	0	0
Tangara	0	0	X	0	0	X	X
Roposeum	0	0	X	0	X	0	X
Lode Enhancer	X	X	0	0	0	0	0
LLMs for Educational games	X	0	0	X	0	0	X
Germinate	X	X	X	X	0	0	0
Morai Maker	0	0	X	X	X	0	0

Table 2
MIAI Component Table

playtest their designs in all conditions, an interesting distinction emerges: Conditions 1–3 followed a linear design process, guiding participants step-by-step. In contrast, Conditions 4 and 5 introduced an interactive loop, allowing participants to design, test, and iterate. Understanding how this interactive structure influences user experience is key to improving MIAI system development.

The aim of this exploratory study was not to define a fixed method for implementing MIAI interfaces, but to explore and contextualize the key factors and features important in their design.

Across all comparisons, participants clearly recognized the value of MIAI systems. When reflecting on their experiences—particularly when comparing the initial condition to the full Melete system—they identified specific advantages that MIAI tools bring to the design process.

7. Declaration on Generative AI

During the preparation of this work, authors used ChatGPT and Grammarly for the following: **grammar and spellcheck**. After using this tool/service, the authors, reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] E. Brynjolfsson, A. McAfee, The second machine age: Work, progress, and prosperity in a time of brilliant technologies, WW Norton & company, 2014.

[2] J. P. Gee, Video games and embodiment, Games and culture 3 (2008) 253–263.

[3] A. Marchand, T. Hennig-Thurau, Value creation in the video game industry: Industry economics, consumer benefits, and research opportunities, Journal of interactive marketing 27 (2013) 141–157.
[4] P. Zackariasson, T. L. Wilson, The video game industry: Formation, present state, and future, Routledge, 2012.
[5] R. D. Gaina, A. Couëtoux, D. J. Soemers, M. H. Winands, T. Vodopivec, F. Kirchgeßner, J. Liu, S. M. Lucas, D. Perez-Liebana, The 2016 two-player gvgai competition, IEEE Transactions on Games 10 (2017) 209–220.
[6] A. Fadaeddini, B. Majidi, M. Eshghi, A case study of generative adversarial networks for procedural synthesis of original textures in video games, in: 2018 2nd National and 1st International Digital Games Research Conference: Trends, Technologies, and Applications (DGRC), IEEE, 2018, pp. 118–122.
[7] J. B. Rocha, R. Prada, Procedural content generation for cooperative games-a systematic review, IEEE Transactions on Games (2025).
[8] R. Ribeiro, A. V. de Carvalho, N. B. Rodrigues, Image-based video game asset generation and evaluation using deep learning: a systematic review of methods and applications, IEEE Transactions on Games (2024).
[9] A. I. Alhussain, A. M. Azmi, Automatic story generation: A survey of approaches, ACM Computing Surveys (CSUR) 54 (2021) 1–38.
[10] K. Fukaya, D. Daylamani-Zad, H. Agius, Intelligent generation of graphical game assets: A conceptual framework and systematic review of the state of the art, ACM Computing Surveys 57 (2025) 1–38.
[11] C. Tollola, Procedural animations in interactive art experiences–a state of the art review, arXiv preprint arXiv:2105.09153 (2021).
[12] M. Božić, M. Horvat, A survey of deep learning audio generation methods, arXiv preprint arXiv:2406.00146 (2024).
[13] J. Togelius, G. N. Yannakakis, K. O. Stanley, C. Browne, Search-based procedural content generation: A taxonomy and survey, IEEE Transactions on Computational Intelligence and AI in Games 3 (2011) 172–186.
[14] J. O'Sullivan, Towards a Digital Poetics: Electronic Literature & Literary Games, Springer, 2019.
[15] M. Barton, S. Stacks, Dungeons and Desktops: The History of Computer Role-Playing Games 2e, CRC Press, 2019.
[16] M. Hendrikx, S. Meijer, J. Van Der Velden, A. Iosup, Procedural content generation for games: A survey, ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM) 9 (2013) 1–22.
[17] M. Guzdial, N. Liao, M. Riedl, Co-creative level design via machine learning, arXiv preprint arXiv:1809.09420 (2018).
[18] C. Brom, J. Gemrot, M. Bida, O. Burkert, S. J. Partington, J. J. Bryson, Posh tools for game agent development by students and non-programmers, in: The Nineth International Computer Games Conference: AI, Mobile, Educational and Serious Games, 2006, pp. 126–133.
[19] E. Giacomello, P. L. Lanzi, D. Loiacono, Doom level generation using generative adversarial networks, in: 2018 IEEE Games, Entertainment, Media Conference (GEM), IEEE, 2018, pp. 316–323.
[20] N. Shaker, M. Shaker, J. Togelius, Ropossum: An authoring tool for designing, optimizing and solving cut the rope levels, in: Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, volume 9, 2013, pp. 215–216.
[21] H. Yu, Reflection on whether chat gpt should be banned by academia from the perspective of education and teaching, Frontiers in Psychology 14 (2023) 1181712.
[22] J. W.-R. M. S. M. Y.-K. Murturi, Sokol, M. Gillies., Melete: Playtesting and 3d environments for mixed-initiative artificial intelligence as a method for prototyping video game levels, CHI Play NA (2025) NA.
[23] A. Baldwin, S. Dahlskog, J. M. Font, J. Holmberg, Towards pattern-based mixed-initiative dungeon generation 20506 (2017) 1–10. doi:10.1145/3102071.3110572.
[24] A. Alvarez, S. Dahlskog, J. Font, J. Holmberg, S. Johansson, Assessing aesthetic criteria in the

evolutionary dungeon designer, in: Proceedings of the 13th International Conference on the Foundations of Digital Games - FDG '18, ACM Press, New York, New York, USA, 2018, pp. 1–4. URL: http://dl.acm.org/citation.cfm?doid=3235765.3235810. doi:10.1145/3235765.3235810.

[25] A. Alvarez, S. Dahlskog, J. Font, J. Holmberg, C. Nolasco, A. Österman, Fostering creativity in the mixed-initiative evolutionary dungeon designer, in: Proceedings of the 13th International Conference on the Foundations of Digital Games - FDG '18, ACM Press, New York, New York, USA, 2018, pp. 1–8. URL: http://dl.acm.org/citation.cfm?doid=3235765.3235815. doi:10.1145/3235765.3235815.

[26] S. Dahlskog, J. Togelius, Patterns, Dungeons and Generators, Foundations of Digital Games (2015).

[27] A. Baldwin, S. Dahlskog, J. M. Font, J. Holmberg, Mixed-initiative procedural generation of dungeons using game design patterns, 2017 IEEE Conference on Computational Intelligence and Games, CIG 2017 (2017) 25–32. doi:10.1109/CIG.2017.8080411.

[28] A. Alvarez, S. Dahlskog, J. Font, J. Togelius, Empowering Quality Diversity in Dungeon Design with Interactive Constrained MAP-Elites, in: 2019 IEEE Conference on Games (CoG), IEEE, 2019, pp. 1–8. URL: https://ieeexplore.ieee.org/document/8848022/. doi:10.1109/CIG.2019.8848022. arXiv:1906.05175.

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

[30] R. Kremers, Level design: concept, theory, and practice, CRC Press, 2009.

[31] D. Wahl, The art of game design: A book of lenses, by jesse schell, The Objective Standard 11 (2016).

[32] C. W. Totten, Level design: Processes and experiences, CRC Press, 2017.

[33] R. Yang, The Level Design Book, volume 1, Self-Published, 2025. URL: https://book.leveldesignbook.com.

[34] K. Salen, E. Zimmerman, This is not a game: Play in cultural environments, in: Proceedings of DiGRA 2003 Conference: Level Up, 2003.

[35] C. Gingold, Miniature Gardens & Magic Crayons: Games, Spaces & Worlds, Ph.D. thesis, School of Literature, Communication, and Culture, Georgia Institute of ..., 2003.

[36] N. Shaker, J. Togelius, M. J. Nelson, Procedural content generation in games (2016).

[37] M. Gillies, R. Fiebrink, A. Tanaka, J. Garcia, F. Bevilacqua, A. Heloir, F. Nunnari, W. Mackay, S. Amershi, B. Lee, et al., Human-centred machine learning, in: Proceedings of the 2016 CHI conference extended abstracts on human factors in computing systems, 2016, pp. 3558–3565.

[38] G. Volkmar, D. Alexandrovsky, A. E. Eilks, D. Queck, M. Herrlich, R. Malaka, Effects of pcg on creativity in playful city-building environments in vr, Proceedings of the ACM on Human-Computer Interaction 6 (2022) 1–20.

[39] ???? URL: https://store.speedtree.com.

[40] A. Chang, Between plants and polygons: Speedtrees and an even speedier history of digital morphogenesis, electronic book review (2019).

[41] A. Liapis, G. N. Yannakakis, J. Togelius, Sentient sketchbook: Computer-aided game level authoring., in: FDG, 2013, pp. 213–220.

[42] G. Smith, J. Whitehead, M. Mateas, Tanagra: A mixed-initiative level design tool, in: Proceedings of the Fifth International Conference on the Foundations of Digital Games, ACM, 2010, pp. 209–216.

[43] M. Guzdial, D. Acharya, M. Kreminski, M. Cook, M. Eladhari, A. Liapis, A. Sullivan, Tabletop roleplaying games as procedural content generators, in: Proceedings of the 15th International Conference on the Foundations of Digital Games, 2020, pp. 1–9.

[44] T. Lubart, How can computers be partners in the creative process: classification and commentary on the special issue, International journal of human-computer studies 63 (2005) 365–369.

[45] Y. Liu, H. Duan, W. Cai, User-generated content and editors in games: A comprehensive survey, arXiv preprint arXiv:2412.13743 (2024).

[46] C. Aliaga, C. Vidal, G. K. Sepulveda, N. Romero, F. González, N. A. Barriga, Level building sidekick: an ai-assisted level editor package for unity, in: Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, volume 19, 2023, pp. 392–399.

[47] A. Liapis, G. Smith, N. Shaker, Mixed-initiative content creation, Procedural content generation in

games (2016) 195–214.
[48] N. Shaker, J. Togelius, M. J. Nelson, N. Shaker, A. Liapis, J. Togelius, R. Lopes, R. Bidarra, Constructive generation methods for dungeons and levels, Procedural Content Generation in Games (2016) 31–55.
[49] A. Alvarez, T. Pederson, J. Salo, C. O. Alm, Invisible AI-driven HCI Systems – When , Why and How (2020).
[50] S. Deterding, M. Akten, K. Compton, R. Fiebrink, M. Gillies, J. Gow, J. Hook, A. Liapis, G. Smith, The chi'17 workshop on mixed-initiative creative interfaces, 2017. URL: https://mici2017.org/, accessed on 23 November 2020.
[51] I. Mele, C. I. Muntean, M. Aliannejadi, N. Voskarides, Workshop on mixed-initiative conversational systems 2021, 2021. URL: https://micros2021.github.io/, accessed on 23 November 2020.
[52] J. R. Carbonell, Mixed-Initiative Man-Computer Instructional Dialogues, Ph.D. thesis, 1970.
[53] R. Dawkins, The Blind Watchmaker, 1986.
[54] S. Todd, W. Latham, Evolutionary Art and Computers, Academic Press Inc, 1992.
[55] G. Lai, F. F. Leymarie, W. Latham, On mixed-initiative content creation for video games, IEEE Transactions on Games 14 (2022) 543–557.
[56] A. Liapis, G. Smith, N. Shaker, Mixed-initiative content creation, in: J. Togelius, N. Shaker, M. J. Nelson (Eds.), Procedural Content Generation in Games: A Textbook and an Overview of Current Research, 2016, pp. 195–214. URL: http://link.springer.com/10.1007/978-3-319-42716-4{_}11. doi:10.1007/978-3-319-42716-4_11.
[57] S. Amershi, M. Cakmak, W. B. Knox, T. Kulesza, Power to the people: The role of humans in interactive machine learning, AI magazine 35 (2014) 105–120.
[58] G. Lai, W. Latham, F. F. Leymarie, Towards friendly mixed initiative procedural content generation: Three pillars of industry, in: International Conference on the Foundations of Digital Games, 2020, pp. 1–4.
[59] A. Drachen, P. Mirza-Babaei, L. E. Nacke, Games user research, Oxford University Press, 2018.
[60] A. Liapis, G. N. Yannakakis, J. Togelius, Sentient sketchbook: Computer-aided game level authoring, Proceedings of the 8th International Conference on the Foundations of Digital Games (FDG 2013) (2013) 213–220. URL: http://www.itu.dk/people/anli/mixedinitiative/sentient{_}sketchbook.pdf{%}5Cnhttp://www.fdg2013.org/program/papers/paper28{_}liapis{_}etal.pdf.
[61] A. Liapis, G. N. Yannakakis, J. Togelius, Designer modeling for Sentient Sketchbook, in: 2014 IEEE Conference on Computational Intelligence and Games, IEEE, 2014, pp. 1–8. URL: http://ieeexplore.ieee.org/document/6932873/. doi:10.1109/CIG.2014.6932873.
[62] N. Shaker, M. Shaker, J. Togelius, Ropossum: An Authoring Tool for Designing, Optimizing and Solving Cut the Rope Levels, Aiide (2013) 215–216.
[63] D. Bhaumik, J. Togelius, G. N. Yannakakis, A. Khalifa, Lode enhancer: Level co-creation through scaling, in: Proceedings of the 18th International Conference on the Foundations of Digital Games, 2023, pp. 1–8.
[64] V. Kumaran, D. Carpenter, J. Rowe, B. Mott, J. Lester, Procedural level generation in educational games from natural language instruction, IEEE Transactions on Games (2024).
[65] M. Kreminski, M. Dickinson, J. Osborn, A. Summerville, M. Mateas, N. Wardrip-Fruin, Germinate: A mixed-initiative casual creator for rhetorical games, in: Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, volume 16, 2020, pp. 102–108.
[66] Z. Zhou, M. Guzdial, Toward co-creative dungeon generation via transfer learning, in: Proceedings of the 16th International Conference on the Foundations of Digital Games, 2021, pp. 1–9.
[67] V. Clarke, V. Braun, N. Hayfield, Thematic analysis, Qualitative psychology: A practical guide to research methods (2015) 222–248.
[68] J. M. Carroll, HCI models, theories, and frameworks: Toward a multidisciplinary science, Elsevier, 2003.
[69] M. Y. BALA, K. DAMLA, A review of human-computer interaction design approaches towards information systems development, BRAIN. Broad Research in Artificial Intelligence and Neuroscience 12 (2021) 229–250.

A. Appendix

A.1. Quantitative Data gathered

Questions	
Q1	How much did you enjoy your experience with the system?
Q2	How successful were you with the system?
Q3	To what extent were you able to use the system?
Q4	How helpful was the system in the design process?
Q5	Is the information provided by the system clear?
Q6	Was the system unclear at any point during your experience?
Q7	Did you feel disorientated or confusion during your participation with the system?
Q8	Do you think that this system will be helpful for you in a development environment?
Q9	Did you find the task difficult?
Q10	Did you find the devices of the system difficult to use?

Table 3
List of Quantitative Questions Asked

Condition Assignment	Condition 1 to 4	Melete
Participant 1	1(Level Editor)	5
Participant 2	2(PCG Output)	5
Participant 3	1(Level Editor)	5
Participant 4	3(Kernal Only)	5
Participant 5	4(MIAI without)	5
Participant 6	1(Level Editor)	5
Participant 7	3(Kernal Only)	5
Participant 8	4(MIAI without)	5
Participant 9	4(MIAI without)	5
Participant 10	2(PCG Output)	5
Participant 11	3(Kernal Only)	5
Participant 12	4(MIAI without)	5

Table 4
Participants Random assignment

ID	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
1	4	4	5	4	4	3	3	4	1	3
3	4	5	4	3	3	4	1	4	1	1
2	5	4	5	5	5	1	1	5	1	1
4	4	3	3	4	2	4	2	5	3	2
5	5	4	4	5	4	1	1	5	1	1
6	5	3	3	2	3	4	4	5	3	3
7	3	2	3	4	5	3	4	2	4	1
8	5	4	4	5	3	2	2	5	1	1
9	4	4	5	4	4	2	2	4	1	1
10	5	5	5	5	5	1	2	5	1	1
11	2	NA	2	2	2	NA	4	3	4	4
12	5	4	4	4	4	4	4	4	4	2

Table 5
Participant responses to Other conditions than Melete

ID	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
1	5	4	5	5	4	3	3	5	2	3
2	5	4	4	5	4	2	1	5	1	1
3	4	5	4	5	3	4	2	4	1	2
4	4	4	4	3	4	3	2	4	2	2
5	5	4	4	4	3	2	1	5	1	1
6	5	4	4	4	3	4	4	5	3	4
7	4	5	5	5	4	3	4	3	1	2
8	5	4	5	5	4	1	2	4	1	1
9	4	4	4	4	4	2	1	4	1	1
10	5	5	5	5	5	1	2	5	2	1
11	3	NA	2	2	1	NA	4	4	3	4
12	4	2	3	4	4	2	5	4	5	5

Table 6
Participant responses to Melete