TÉCNICO LISBOA

Personality-based Dialogues for Bystander Non-Player Characters

João Alexandre Respeita Barbosa

Thesis to obtain the Master of Science Degree in

Computer Science and Engineering

Supervisor: Prof. Carlos Martinho

October 2025

Declaration
I declare that this document is an original work of my own authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

In the preparation of this thesis, AI tools were employed in a supporting capacity. ChatGPT was primarily used for language refinement and drafting assistance, Claude Sonnet 4 provided help with LaTeX structure and figures, and DeepSeek-Chat V3 was integrated during the system implementation.

Acknowledgments

I would like to express my deepest gratitude to everyone who supported and accompanied me throughout this academic and personal journey.

To my family - Your unwavering love, patience and encouragement have been the foundation of everything I have achieved. Thank you for always believing in me and for giving me the strength to move forward, even in the most challenging moments. For being there unconditionally and raising me to be a better person: my mother Paula, my brother Nuno, my sister Joana and my father Paulo — this achievement is as much yours as it is mine.

To my lifelong friends outside the academic world, thank you for being a constant source of laughter, perspective, and emotional support. You made the hardest days bearable and the good times unforgettable. Duarte, Carita, Babá, Pinto, Olavo, Durão, Mateus, Rocha, Ricardo... you've each contributed to this journey in your own way, and I carry a piece of you in everything I do.

To my advisor, Professor Carlos Martinho, thank you for your guidance, trust, and thoughtful feedback throughout this thesis. Your knowledge, clarity and consistent availability were instrumental, and I consider myself extremely lucky to have had your support.

To my colleagues and classmates, especially those who shared this academic path with me, your companionship, teamwork, and motivation made even the toughest challenges feel achievable. Lima, Inês, Batalheiro, Rique, and Zé, you are not just colleagues; you are good friends that this university has gifted me.

Finally, a sincere thank you to everyone who participated in the testing phase of this thesis. Your time, feedback and insights were crucial to the development of this project and are truly appreciated.

To all of you,thank you for being part of this chapter in my life.

Abstract

This thesis explores the application of Large Language Models (LLMs) to support the creation of coherent, natural, and personality-driven dialogues for Non-Playable Characters (NPCs) in visual novel games. A key contribution is the integration of an LLM into the game editor to assist designers in generating and refining NPC conversations at editor-time in a human-in-the-loop workflow.

To enhance realism, each NPC is defined by a distinct personality profile, using the Five-Factor Model (FFM), which influences their dialogue style and interaction patterns. Additional customizable parameters further refine and guide the generation process. Rather than replacing the writer, the LLM acts as a creative assistant, suggesting dialogue lines aligned with the personalities and context defined by the user.

The aim of this work is to streamline the writing process for developers while improving the believability and narrative quality of NPC interactions. To assess the effectiveness of this approach, a user testing phase was conducted to evaluate dialogue realism, personality coherence, and the usefulness of LLM-generated suggestions in the design process. This work aspires to support the development of more immersive storytelling experiences and to demonstrate the practical value of Artificial Intelligence (AI)-assisted tools in game narrative design.

Keywords

Large Language Models, Artificial Intelligence, Non-Player Characters, Personality Traits, Five-Factor Model, Visual Novel Game.

Resumo

Esta tese explora a aplicação de Large Language Models (LLMs) como apoio à criação de diálogos mais coerentes, naturais e orientados pela personalidade dos Non-Playable Characters (NPCs) em jogos visual novel. Uma das principais contribuições deste trabalho é a integração de um LLM no editor do jogo, especificamente na interface de edição de diálogos, para auxiliar os desenvolvedores na geração e refinamento de conversas entre NPCs na fase de desenvolvimento.

Para aumentar o realismo, cada NPC é definido por um perfil de personalidade distinto, usando o Five-Factor Model (FFM), o qual influencia o estilo dos seus diálogos e os seus padrões de interação. Parâmetros adicionais e personalizáveis ajudam ainda a orientar o processo de geração como vai ser abordado mais à frente. Em vez de substituir o escritor, o LLM atua como um assistente criativo, sugerindo linhas de diálogo alinhadas com as personalidades e o contexto definidos pelo utilizador.

O objetivo deste trabalho é facilitar o processo de escrita para os desenvolvedores, ao mesmo tempo que contribui para o aumento do realismo e da qualidade narrativa das interações entre NPCs. Para avaliar a eficácia desta abordagem, foi conduzida uma fase de testes com utilizadores, com o intuito de analisar o realismo dos diálogos, a coerência das personalidades e a utilidade das sugestões geradas pelo LLM no processo de design. Este trabalho procura apoiar o desenvolvimento de experiências narrativas mais imersivas e demonstrar o valor prático de ferramentas assistidas por Artificial Intelligence (AI) no design narrativo de jogos.

Palavras Chave

Modelos de Linguagem de Grande Escala, Inteligência Artificial, Traços de Personalidade, Modelo dos Cinco Grandes Fatores, Desenvolvimento de Jogos, Geração de Diálogo.

Contents

List of Figures

List of Tables

Listings

Acronyms

RPG	Role-Playing Game
NPC	Non-Playable Character
LLM	Large Language Model
AI	Artificial Intelligence
GPT	Generative Pre-training Transformer
NLP	natural language processing
GDPR	General Data Protection Regulation
HCI	Human Computer Interaction
IQR	interquartile range
FFM	Five-Factor Model
MBTI	Myers–Briggs Type Indicator
API	Application Program Interface
UI	User Interface

1

Introduction

Contents

1.1 Motivation and Context

Artificial Intelligence (AI) has been an integral part of video games since their inception in 1952, first seen in the game Nim [1]. As the years go by, both its popularity and evolution have increased, becoming a pivotal element in video game development, enabling groundbreaking advancements in gameplay and player immersion [2]. For instance, in The Last of Us Part II, AI is used to create Non-Playable Characters (NPCs) with realistic behaviors, such as reacting dynamically to the player's actions and coordinating as a team for a combat fight [3]. Similarly, No Man's Sky employs procedural generation powered by AI to create an expansive universe of planets, each with unique ecosystems and terrains [4]. These examples demonstrate how AI is not only enhancing technical aspects of games but also reshaping storytelling and emotional engagement, offering richer and more immersive gaming experiences.

The development of NPCs in video games has traditionally relied on prescripted dialogue trees or rule-based systems, which often lack the flexibility and realism necessary for an immersive gameplay from the player's point of view, and can also be a headache from the game designer's perspective [5]. Recent advancements in Large Language Models (LLMs), such as Generative Pre-training Transformer (GPT)-based architectures, have revolutionized the field by enabling the generation of dynamic, context-aware, and naturalistic dialogues. These models can adapt to various conversational scenarios and personality traits, offering a significant leap in creating NPCs that feel more lifelike and engaging. By leveraging LLMs, game developers can design characters with unique personalities and behaviors, enhancing player interaction and narrative depth [6]. This approach not only enriches the gaming experience but also opens new avenues for educational and socially relevant applications, where realistic NPC interactions can simulate complex human dynamics while facilitating human work.

1. What is the problem?

   Generating dynamic and realistic dialogues remains a significant challenge in game design. This thesis explores the use of LLMs to address these issues by creating NPC dialogues that are adaptive, context-aware, and reflective of individual personalities. While LLMs have shown great promise in generating natural language, their application in this context also raises potential challenges, such as ensuring coherence in conversations, maintaining character consistency, and avoiding inappropriate or unintended outputs [7]. These issues highlight the need for careful design and evaluation to realize the full potential of LLMs in revolutionizing NPC interactions.

2. Why is it interesting and important?

   Dynamic NPC interactions can greatly enhance the realism and depth of games, improving player immersion. For instance, in early games like The Elder Scrolls III: Morrowind (2002), NPCs relied heavily on static text-based dialogues that were often repetitive and lacked personality. While

these NPCs served their functional purpose at the time, they contributed minimally to immersion or emotional connection. In contrast, modern games like Red Dead Redemption 2 (2018) showcase the potential of well-designed NPC dialogue. This evolution demonstrates that NPC dialogue plays a crucial role in fostering emotional investment, shaping the narrative, and making the game world feel alive. By advancing dialogue generation through AI, we aim to push this boundary even further, creating NPCs that offer more dynamic and context-aware interactions [8]. This approach has the potential to advance the entertainment value of games.

3. Why is it hard?

   Generating realistic dialogues dynamically is inherently challenging due to the complexity of human communication. Conversations are nuanced, context-dependent, and shaped by individual personality traits, cultural influences, and social cues. Even in controlled environments like movies, where scripts are crafted by professional writers, creating realistic and emotionally resonant dialogue is a notoriously difficult task. Translating that effort into a video game setting, where dialogues must adapt dynamically to unpredictable player interactions, introduces an even greater challenge.

4. Why hasn't it been solved before?

   Previous attempts to automate NPC dialogue generation have faced challenges in balancing quality, variability, and efficiency. Traditional methods like scripted or rule-based approaches demand extensive manual effort and struggle to adapt to diverse character personalities or dynamic contexts. While LLMs offer potential for generating natural-sounding text, their integration has primarily been explored in runtime applications, rather than tools for assisting designers during development, more specifically as a framework within the game editor for dialogues. Additionally, concerns like controlling AI output and aligning it with specific narrative goals have limited adoption. This approach balances AI assistance with human oversight, ensuring quality and thematic appropriateness without relying on unpredictable runtime generation.

5. What are the key components of my approach and results?

   The approach integrates an LLM into the game editor to assist designers at editor-time, generating personality, and context-aligned NPC dialogue in a human-in-the-loop workflow. Designers configure traits, relationships, and scene parameters; the system proposes editable suggestions that preserve creative control. This system leverages the capabilities of LLMs to produce high-quality, pre-generated dialogue options tailored to character profiles and game scenarios, such as conversations between students, each with their own identity. The results aim to demonstrate how this integration can save time, improve narrative consistency, and support the creative process in game development. However, certain limitations exist. LLMs can produce outputs that require careful review to ensure appropriateness and alignment with the game's narrative goals.

1.2 Work Objectives

The primary objective of this work is to integrate an LLM into a game editor to assist game designers in generating personality-driven, contextually relevant NPC dialogues during the development phase. This system will aim to streamline the dialogue creation process by offering dynamic, high-quality dialogue suggestions that align with the desired character personalities and game scenarios. By embedding this functionality within the editor, the project aims to empower designers to efficiently create engaging and consistent NPC interactions while maintaining full creative control over the final output.

This work addresses two key research questions: **(1)** Can LLMs effectively generate contextually appropriate and character-aligned dialogue content? **(2)** What are the key factors for successful implementation of LLM-based dialogue generation systems in game development workflows?

1.3 Expected Contributions

This project is expected to contribute to the fields of game development and AI by demonstrating how LLMs can be seamlessly integrated into game design tools to enhance the workflow for creating NPC dialogues. It will provide a practical framework for leveraging AI into the development pipeline, offering a balance between automated generation and human creativity. By focusing on generating contextually appropriate and personality-driven dialogues for students, this work highlights the potential of LLMs to streamline narrative creation while maintaining high-quality interactions. Additionally, the project will emphasize how integrating AI into development tools can improve productivity for designers and support the creation of more immersive and engaging player experiences.

1.4 Thesis Outline

This thesis is organized into six chapters that systematically address the integration of LLMs into game development workflows for enhanced NPC dialogue generation:

Chapter 2 - Related Work surveys the theoretical foundation across key areas including personality-driven NPC design using psychological frameworks, evaluation methods for dialogue quality, dialogue systems evolution in video games, current applications of LLMs in narrative generation, and existing game development tools. The chapter identifies gaps in current approaches and establishes how this thesis addresses limitations in character modeling depth and AI-assisted dialogue creation.

Chapter 3 - Implementation describes the system architecture and technical implementation of the LLM-integrated dialogue generation tool. The chapter presents the human-in-the-loop workflow design, defines functional and non-functional requirements, details the modular system architecture with core

components, and explains the integration with the existing Ren'Py-based game editor.

Chapter 4 - Experiment presents the experimental design and methodology employed to evaluate the system. The chapter outlines the mixed-methods approach, details the study design with 12 participants, describes the testing protocol using a Portuguese high school scenario, and defines the evaluation metrics and measurement frameworks for assessing dialogue generation quality, usability, and user satisfaction.

Chapter 5 - Results analyzes the data collected during the evaluation study. The chapter presents participant characteristics, reports quantitative and qualitative findings on system usability, AI-generated content quality, and user satisfaction, discusses implications for AI-assisted creative workflows, and addresses study limitations and potential biases.

Chapter 6 - Conclusion and Future Work synthesizes the research findings and contributions. The chapter summarizes the thesis achievements, revisits the research questions with definitive answers, outlines key technical and empirical contributions, acknowledges system limitations, and proposes future research directions including expanded parameters, and broader creative applications.

2

Related Work

Contents

2.1 Personality-Driven NPC Design

Psychology plays a vital role in understanding human behavior, communication and social interaction, all of which are essential for designing realistic NPC dialogues. The integration of psychological principles in game design has long contributed to the creation of immersive and believable NPCs [9]. Core psychological concepts, such as personality traits, emotional responses and interpersonal dynamics, are crucial for crafting believable characters that can convincingly portray different roles in a high school setting.

Frameworks such as the Five-Factor Model (FFM) of personality provide a structured way to define traits like openness, conscientiousness, extraversion, agreeableness, and neuroticism [10, 11]. These traits can be applied to guide both the behavior and the dialogue of NPCs, helping to ensure consistency and depth in their interactions. For example, an NPC with high agreeableness may speak in a cooperative and empathetic tone when acting as a supportive bystander, while one with high neuroticism might respond anxiously when witnessing bullying situations.

Personality-driven NPCs have been explored extensively in behavior modeling for virtual environments, including digital games [12, 13]. In *F.E.A.R.* (First Encounter Assault Recon)[1], NPCs exhibit coordinated team-based behaviors that are influenced by predefined personality traits. *The Sims*[2] franchise [14] features NPCs whose actions and reactions are shaped by traits like extraversion or neatness, demonstrating how personality traits can drive emergent storytelling with NPCs reacting dynamically to both their characteristics and environmental stimuli. Similarly, *Persona 5*[3] integrates social dynamics into its gameplay loop, where NPC behaviors and relationships influence the player's progression, notably, the entire *Persona* series is itself grounded in Carl Jung's psychological theories, particularly the concept of personas, psychological archetypes and the collective unconscious [15], demonstrating how deeply psychological principles can be embedded into game design to create meaningful character interactions and narrative experiences.

Several research projects have adapted psychological frameworks to influence NPC behaviors in serious games and educational simulations. For instance, *NPCs as People, Too: The Extreme AI Personality Engine* by Georgeson and Child (2016) [16] explores adaptive NPC personalities based on all thirty facets of the FFM. Recent research by Han (2025) [17] investigated the capability of LLMs to generate behaviors for embodied virtual agents based on personality traits, demonstrating promising results in translating psychological constructs into agent behaviors through language model conditioning. Further supporting this line of research, Sorokovikova (2024) [18] provided additional evidence that LLMs can effectively simulate the FFM personality traits, reinforcing the theoretical foundation for using language models in personality-driven character systems.

[1] *F.E.A.R.*, developed by Monolith Productions and Wargaming Chicago-Baltimore, first released in 2005.
[2] *The Sims franchise*, developed by Maxis, first released in 2000.
[3] *Persona 5*, developed by Atlus, first released in 2016.

A – Justification of the personality model The FFM was chosen as the personality framework for this thesis because it benefits from extensive empirical support and demonstrated predictive validity across a wide range of behavioral and outcome measures [19, 20]. The FFM describes personality as continuous dimensions, which facilitates numerical parametrization and fine-grained variation when configuring NPCs [21]. For practical usability within the editor, these continuous trait scores are exposed to designers as three discrete levels (low, medium, high). This discretization simplifies the authoring workflow while preserving the underlying dimensional semantics; internally the system can retain continuous representations and map them to behavioral parameters as needed. By contrast, typological schemes like the Myers–Briggs Type Indicator (MBTI), that categorizes individuals into one of 16 personality types, have been criticized for lower reliability and weaker empirical foundations [22]. Alternative trait models, notably HEXACO, add an honesty–humility factor and can improve prediction for specific behaviors, but such improvements tend to be domain-specific rather than a general replacement for the FFM [23]. For these reasons the FFM represents a pragmatic and well-documented choice for conditioning character behavior and informing prompt-based personality descriptions in the dialogue generation system developed here; the limitations and domain-dependent advantages of other models are acknowledged and discussed.

This thesis takes an alternative approach, leveraging psychological insights through the use of an LLM to assist game designers during dialogue creation. By embedding personality traits into the LLM's prompts or conditioning data, it becomes possible to generate dialogue that reflects psychological principles while reducing manual workload. However, this also introduces new challenges, such as ensuring that generated content consistently aligns with the intended traits and remains coherent across multiple exchanges.

2.2 Evaluation Methods for NPC Dialogue Quality

Evaluating the quality of NPC dialogues has traditionally relied on a combination of player testing, expert review, and automated metrics [24]. In practice, playtesting remains one of the most effective ways to assess dialogue authenticity, emotional impact, and player engagement. For example, studios such as BioWare use iterative player feedback during the development of games like *Mass Effect*[4], where branching dialogue trees are refined based on how players perceive and respond to different choices.

In academic contexts, studies often adopt structured surveys, interviews, and controlled experiments to evaluate specific aspects of NPC interactions, such as narrative coherence [25], engagement, and emotional resonance [26]. This approach is especially valuable for identifying moments where AI-generated dialogue deviates from expectations or disrupts immersion.

[4] *Mass Effect*, developed by BioWare, first released in 2007.

When games aim to simulate realistic social dynamics, expert evaluation is also employed. For instance, researchers have collaborated with specialists in psychology or sociology to assess the plausibility of in-game interactions, particularly in scenarios involving complex interpersonal relationships or sensitive themes [27]. These expert reviews offer nuanced qualitative feedback, though they require significant time and interdisciplinary collaboration.

The approach in this thesis draws on these established practices by combining feedback from typical players with input from an individual with a background in psychology. This mirrors previous studies that balance narrative and gameplay quality with evaluations of social and behavioral realism, ensuring that NPC dialogues are assessed from both a general audience and a specialist perspective.

2.3 Measuring Immersion, Engagement, and Perceived Realism

The assessment of immersion, engagement, and perceived realism in NPC interactions has been widely studied in the fields of game research and Human Computer Interaction (HCI). Previous works have proposed both subjective and objective measures to capture these dimensions.

2.3.1 Immersion

Immersion is often described as the degree to which players feel "inside" the game world [28, 29]. Common approaches to measuring immersion include:

- **Subjective Scales** — Instruments such as the Immersive Tendencies Questionnaire (ITQ) assess a participant's predisposition to become immersed in virtual environments.
- **Behavioral Observation** — Metrics such as time spent interacting with NPCs or voluntary engagement inside quests provide indirect evidence of immersion levels.

2.3.2 Engagement

Engagement reflects the player's motivation and sustained interest in interacting with game content [30, 31]. Reported measurement strategies include:

- **Interaction Rate** — Quantifying the number of interactions with NPCs during play sessions.
- **Engagement Scales** — Instruments such as the Game Engagement Questionnaire (GEQ) [32] assess emotional investment, interest, and motivation.
- **Qualitative Analysis** — Player feedback can provide insights into whether NPC interactions influenced decision-making or enhanced the overall experience.

2.3.3 Perceived Realism

Perceived realism refers to the extent to which players interpret NPC interactions as believable and consistent with real-life social behavior [33]. This dimension has been measured through:

- **Realism Scales** — For example, the Perceived Realism Scale [34] evaluates the naturalness and believability of dialogues.
- **Character Consistency Analysis** — Assessing whether dialogue aligns with predefined traits such as personality and backstory.
- **Qualitative Interviews** — Gathering player reflections on how in-game conversations compare to real-world exchanges.

These combined approaches form a comprehensive methodological foundation for evaluating NPC dialogues. The methods used in this thesis are informed by these established practices, aiming to capture both experiential and behavioral aspects of a player and a NPC interaction.

2.4 Dialogue Systems in Video Games

Dialogue systems are central to game storytelling, allowing players to interact meaningfully with NPCs and affecting the progression of a game's narrative. Over time these systems have evolved from simple choice menus with predefined responses to more complex and context-sensitive interactions, increasing player immersion and emotional engagement.

Early dialogue systems were mainly text based and followed a rigid branching structure [35]. Players selected from a set of options and the NPC response was predetermined. A classic example is the *Ultima I: The First Age of Darkness*[5], where conversations consisted of a few lines with little variation. These systems were functional but lacked flexibility and depth.

With the rise of Role-Playing Games (RPGs) and open world games, dialogue systems introduced branching narratives that could change the course of a game. Titles such as *The Elder Scrolls V: Skyrim*[6] and *The Witcher 3: Wild Hunt*[7] use dialogue trees in which player choices influence relationships, quests and, in some cases, endings. These approaches offer multiple outcomes, but typically rely on predefined scripts and extensive authoring to cover possible player actions.

More recently there has been a shift towards dynamic and context sensitive dialogue. These systems take into account not only explicit player choices but also the broader game context. For example, *Red Dead Redemption 2*[8] features NPCs that react in real time to the player's behavior. Similarly, the *Mass*

[5] Ultima, developed by Richard Garriott, first released in 1981.
[6] *The Elder Scrolls V: Skyrim*, developed by Bethesda Game Studios, first released in 2011.
[7] *The Witcher 3: Wild Hunt*, developed by CD Projekt RED, first released in 2015.
[8] *Red Dead Redemption 2*, developed by Rockstar Games, first released in 2018.

Effect[9] series implemented alignment mechanics such as “Paragon” and “Renegade”, where actions and dialogue options influence character relationships and narrative consequences.

Procedurally generated dialogue is among the most innovative developments in the field [36]. Instead of relying solely on prewritten text, these systems generate responses from algorithms that combine narrative context, character traits and game state. Early work and recent prototypes suggest that procedural approaches can produce more adaptive and varied conversations, potentially reducing the need for exhaustive branching authoring [37].

Despite these advances, important challenges remain. Common issues include repetitive lines, lack of emotional depth, and occasional mismatches between player actions and NPC responses. A well known player frustration example is the frequent, context poor prompts such as “Hey! Listen!” from Navi in *The Legend of Zelda: Ocarina of Time*[10]. Moreover, creating content to cover many possible interactions requires substantial resources.

2.5 LLMs for Dialogue Generation

The use of LLMs in game design is an active and rapidly growing area of research, with recent advances highlighting their potential to transform how narratives and character dialogues are authored. Models such as OpenAI’s GPT and Google’s Gemini have been applied to generate dynamic, context-aware dialogues that adapt to player interactions [38]. These models, trained on very large text corpora, offer high fluency, stylistic flexibility and the ability to produce varied responses, which makes them attractive for games that require extensive conversational content or non linear storytelling.

Experimental projects demonstrate both the promise and the practical limits of current approaches. For example, AI Dungeon used GPT-3 to create open-ended interactive stories where the narrative evolves directly from player input, producing rich and surprising outcomes but also exposing issues such as topical drift and incoherent long term structure. More recent demonstrations, such as *The Matrix Awakens*[11], showcase how AI can generate more lifelike NPC behavior in constrained environments, with conversations that appear more fluid and less scripted than classical branching systems.

Despite these advances, integrating LLMs into game production raises several technical and design challenges. Maintaining narrative consistency across long, multi-turn interactions is difficult because models may contradict prior statements or lose track of goals. Models can generate irrelevant or unsafe content, and they can reflect biases present in training data. Deploying large models at runtime also introduces cost and latency concerns that affect responsiveness and scalability. In addition, controlling

[9] *Mass Effect*, developed by BioWare, first released in 2007.
[10] *The Legend of Zelda: Ocarina of Time*, developed by Nintendo, first released in 1998.
[11] *The Matrix Awakens*, developed by Epic Games, first released in 2021. For more information, visit: https://www.unrealengine.com/en-US/blog/introducing-the-matrix-awakens-an-unreal-engine-5-experience.

tone and personality at scale requires careful conditioning, as naive prompting can produce inconsistent character behavior.

A fundamental aspect of addressing these challenges is the strategic design and optimization of prompts, the input instructions that guide model behavior and output quality. Prompt engineering has emerged as a critical discipline that significantly influences the effectiveness of LLM applications across diverse domains [39]. In the context of dialogue generation, prompt engineering serves multiple essential functions. Well-crafted prompts can establish character voice consistency by embedding personality traits, emotional states, and behavioral patterns directly into the model's conditioning context. They enable contextual awareness by incorporating scene information, relationship dynamics, and conversation history that inform appropriate response generation. Furthermore, sophisticated prompting strategies can implement quality control mechanisms that guide models toward desired narrative goals while avoiding inappropriate or off-topic content.

A – DeepSeek-V3 is an LLM developed by DeepSeek AI, designed to deliver high-quality natural language understanding and generation across multiple domains. It builds upon transformer-based architectures optimized for efficiency and contextual coherence, offering strong multilingual capabilities and improved control over style, tone, and factual consistency. The model is trained on diverse datasets combining conversational, narrative, and technical text sources, enabling it to perform well in both creative and analytical tasks [40]. Its balance between fluency and controllability makes it suitable for research-oriented applications such as dialogue generation, content authoring, and interactive narrative design. In this project, DeepSeek-V3 was chosen primarily because it offered one of the best-performing freely accessible models available on OpenRouter A –, providing reliable and natural output in European Portuguese, essential requirement for the dialogue generation tasks conducted in this study [41].

The importance of prompt engineering becomes particularly evident when considering the complexity of human dialogue, which depends not only on linguistic competence but also on social awareness, emotional intelligence, and cultural understanding. Effective prompts must translate these multifaceted requirements into structured instructions that LLMs can interpret and apply consistently. This translation process represents a critical bridge between human creative intent and machine-generated content, determining whether AI assistance enhances or hinders the creative workflow. Research and practical applications have demonstrated that prompt engineering can dramatically improve output quality, consistency, and relevance compared to naive or generic prompting approaches [42].

Strategies to address the broader integration challenges have emerged in the literature and in industry practice. The systematic design of prompts, incorporating techniques such as few-shot examples, structured formatting, and explicit constraint specification, enables more predictable and controllable LLM behavior, which is essential for professional creative applications where quality and appropriate-

ness are paramount. Recent work on prompt tuning specifically for dialogue generation has demonstrated significant improvements in output quality and character consistency through systematic prompt optimization approaches [43]. Conditioning on structured character state or short memory buffers can improve local consistency. Retrieval augmented generation and grounding with game data help anchor outputs to world facts and reduce hallucination. Post processing, filtering and safety classifiers are often applied to remove or flag inappropriate content. For more persistent control, teams may fine tune smaller models on curated dialogue corpora or use supervised adapters to align outputs with design constraints. Many studios prefer to apply LLMs during editor-time as drafting tools rather than as fully autonomous runtime agents, trading some generative flexibility for greater editorial control [44].

2.6 Game Development Tools and AI Integration

The process of game development involves a wide range of tools designed to streamline workflows and enable the creation of complex, interactive environments. Tools such as Unity[12], Unreal Engine[13], and GameMaker Studio[14] are widely used for designing mechanics, levels and assets. These platforms provide developers with robust environments for scripting, asset management and testing, enabling the creation of games with varying levels of complexity. While traditionally focused on physics, rendering and game logic, these engines are increasingly incorporating AI technologies to expand creative possibilities for developers.

AI integration in game development tools is becoming a key focus area [44, 45]. For example, Unity now offers an in-Editor AI suite (Unity AI[15]) to support asset creation, in-Editor chat and other generative workflows. Unreal Engine provides MetaHuman Creator[16] to produce high-fidelity digital humans for animation and character pipelines. Tools such as Promethean AI[17] support environment and asset generation by assisting artists in set dressing, asset search and scene assembly, significantly reducing iteration time.

Dialogue generation and narrative design are also benefiting from AI and natural language processing (NLP) advances [46, 47]. Several tools and experimental pipelines assist writers by generating draft dialogue, suggesting variations or grounding lines with in-game facts, which can accelerate editor-time while preserving authorial intent.

[12] Unity, developed by Unity Technologies, for more information visit https://unity.com/.
[13] Unreal Engine, developed by Epic Games, more details are available at https://www.unrealengine.com/.
[14] GameMaker Studio, developed by YoYo Games, visit https://www.yoyogames.com/en/gamemaker for more information.
[15] Unity AI, developed by Unity Technologies, for more information visit https://unity.com/products/ai.
[16] MetaHuman Creator, developed by Epic Games, for more information visit https://dev.epicgames.com/documentation/en-us/metahuman/metahuman-creator.
[17] Promethean AI, developed by Andrew Maximov, for more information visit https://www.prometheanai.com/.

A – OpenRouter Gateway. OpenRouter is a unified API gateway that provides access to multiple large language models through a single, OpenAI-compatible interface. It allows developers to experiment with different models, such as GPT, Claude, Mistral, or DeepSeek, without changing the underlying integration logic. This interoperability makes it particularly valuable for research and development environments where comparative testing or flexible deployment is needed. By centralizing authentication, rate limiting, and billing, OpenRouter simplifies the management of multi-model workflows and facilitates the inclusion of LLMs in creative tools such as game editors. Its modularity also enables developers to quickly adapt to evolving model capabilities, maintaining a stable communication layer independent of specific providers [48].

The integration of AI within these tools is not without challenges. Ensuring that AI generated content aligns with a game's vision, tone and narrative requires oversight and fine tuning. Usability is also critical: tools must be accessible to designers who may not have AI expertise. There are also ethical and practical concerns such as bias in generated content, safety filtering and the computational cost of model inference. Despite these obstacles, the inclusion of AI in game development pipelines has proved to be a powerful enabler, allowing smaller teams to achieve results that previously required larger resources.

2.7 Game and Editor

To demonstrate the practical application of the proposed LLM-assisted dialogue generation system, this thesis focuses on a specific game development context that exemplifies the challenges and opportunities in creating realistic NPC interactions. The following sections present the game "Brave" 2.7.1 and its associated editor 2.7.2, which serve as the primary evaluation platform for assessing the effectiveness of personality-driven dialogue generation in realistic high school scenarios that contain sensitive scenarios such as bullying and cyberbullying situations.

The selection of this particular game and editor combination is motivated by several key factors. The high school setting provides a natural environment for exploring complex social dynamics, including the sensitive topic of cyberbullying that the game is designed to address. The visual novel format allows for deep character interactions and dialogue-heavy gameplay, making it an ideal platform for testing dialogue generation quality and personality consistency. Finally, the game's focus on bystander observation and intervention training aligns directly with the educational objectives of anti-cyberbullying formation tools.

2.7.1 Brave

The game is designed as a visual novel series set in a high school, where the player takes on the role of a student. The player is tasked with various responsibilities such as attending lessons, doing schoolwork and collaborating with other students on favors or challenges. Beyond these structured tasks, the core gameplay element involves exploring the school environment and passively observing interactions among students (figure 2.1). These interactions are often subtle, requiring the player to listen carefully and interpret the context of dialogues (figure 2.2). A critical aspect of the game is to identify and understand situations, such as bullying cases, based on these observed dialogues. After analyzing the scenarios, the player must decide on appropriate actions.

Figure 2.1: Game screen with a dialogue to interact and arrows to navigate through areas

Figure 2.2: Dialogue interaction

2.7.2 Editor

The game editor is specifically designed to support the creation of visual novel-style games, such as the high school game described above. Both the game and the editor are built on Ren'Py (figure 2.3), a Python-based game engine widely used to create interactive stories and visual novels [49]. Ren'Py provides a robust foundation for narrative-driven games, offering a simplified scripting language alongside the flexibility of Python for advanced customizations.

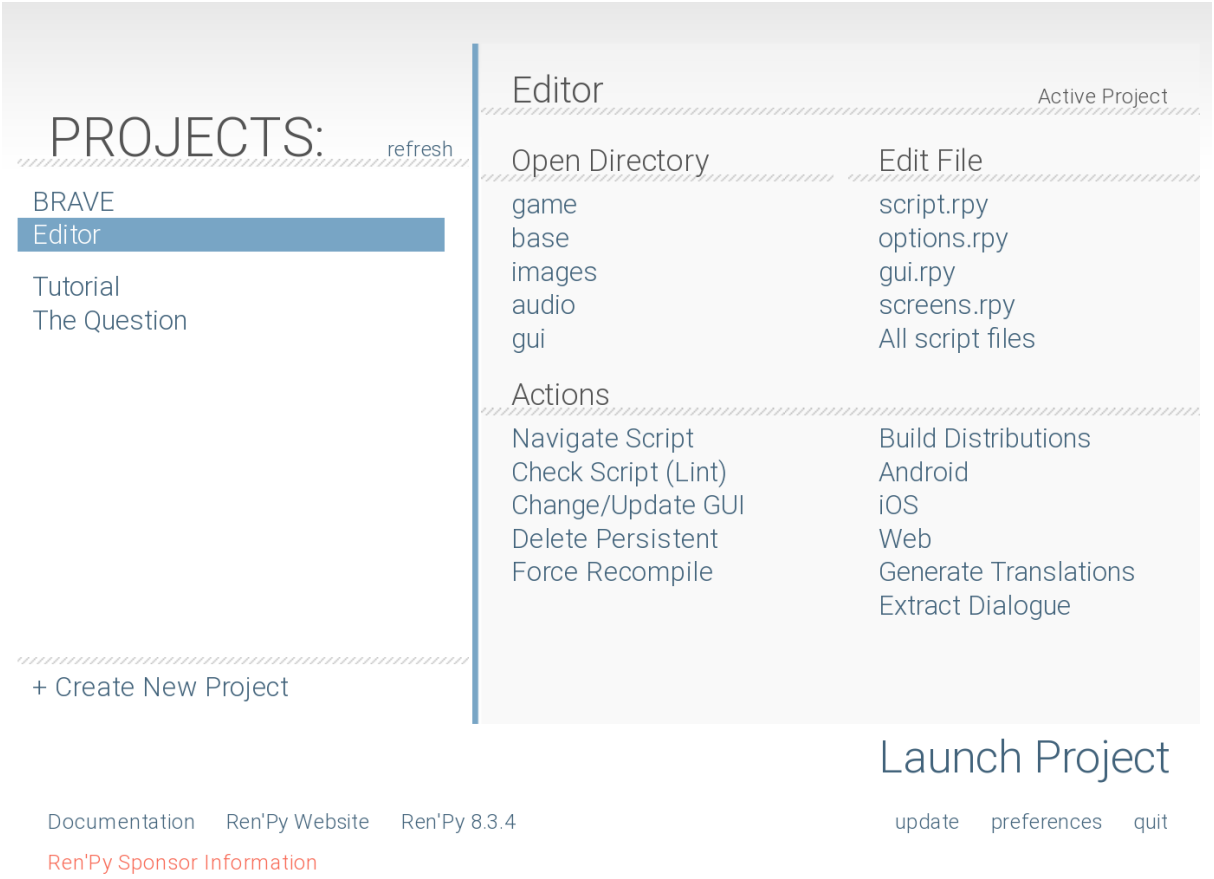


Figure 2.3: Ren'Py starter window with Brave and Editor projects

The editor provides a comprehensive development environment with two main functional areas: **New** and **Library** (figure 2.4). The New section allows developers to create fresh content elements, while the Library section provides access to previously created assets and enables their edition or deletion.

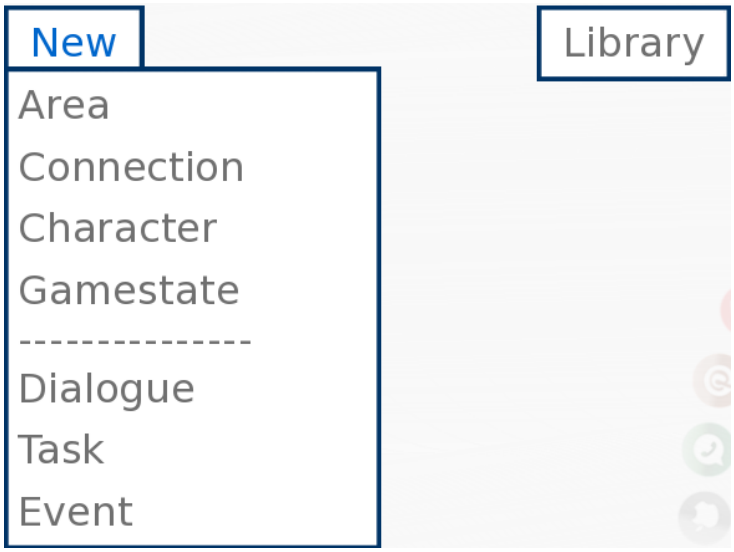


Figure 2.4: Editor starter window with New and Library sections

2.7.2.A Content Creation Capabilities

The editor supports the creation of seven distinct content types (figure 2.4), each serving specific purposes in game development:

Area Management: Developers can design game environments by selecting background images (figure 2.5) and configuring spatial properties. Each area includes visual positioning for character placement and dialogue alignment, with a preview showing how characters will appear in the environment (figure 2.6). Additionally, designers can define the connections between areas, allowing the player to navigate through them.

Character Development: The character creation system allows developers to define basic character properties including name, visual appearance, and color coding for dialogue identification. Visual customization includes portrait selection from a library of diverse character images and body representations for environmental placement (figure 2.7).

Dialogue System: The dialogue creation interface allows developers to construct conversations between multiple characters (figure 2.8). The system supports multi-character conversations with emotional expression selection for each dialogue line, enabling developers to associate specific facial expressions with character speech. Advanced features include dialogue prioritization, trigger conditions, and consequences that affect the game state.

Task and Event Management: Interactive tasks can be positioned within game areas using a visual placement system. Tasks are organized by session and section, with customizable icons and visual properties (figure 2.9). Events provide dynamic game state manipulation, supporting actions such as enabling or disabling areas, triggering specific dialogues, and modifying character accessibility.

Figure 2.5: Area creation with background options

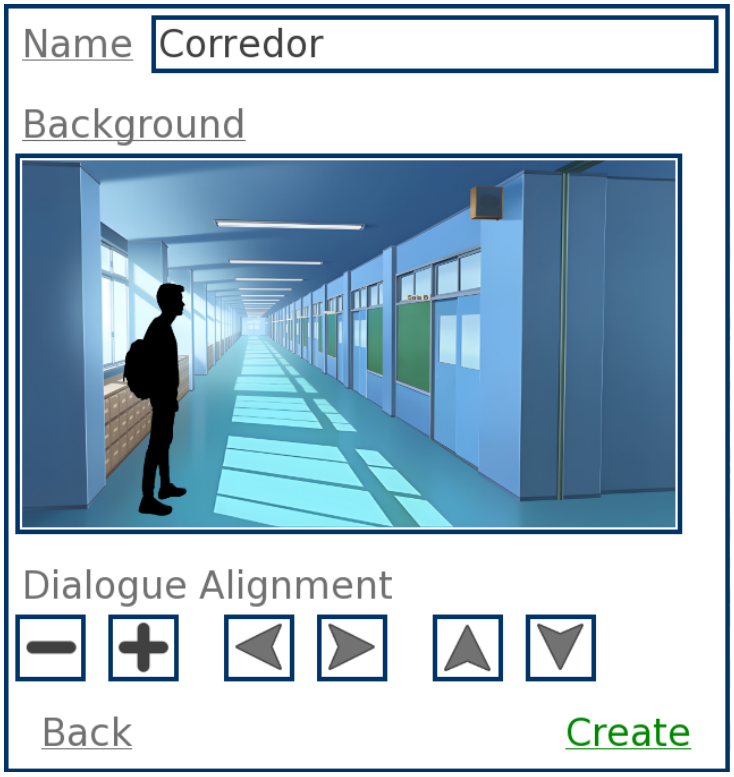


Figure 2.6: Area Creation with character/dialogue placement

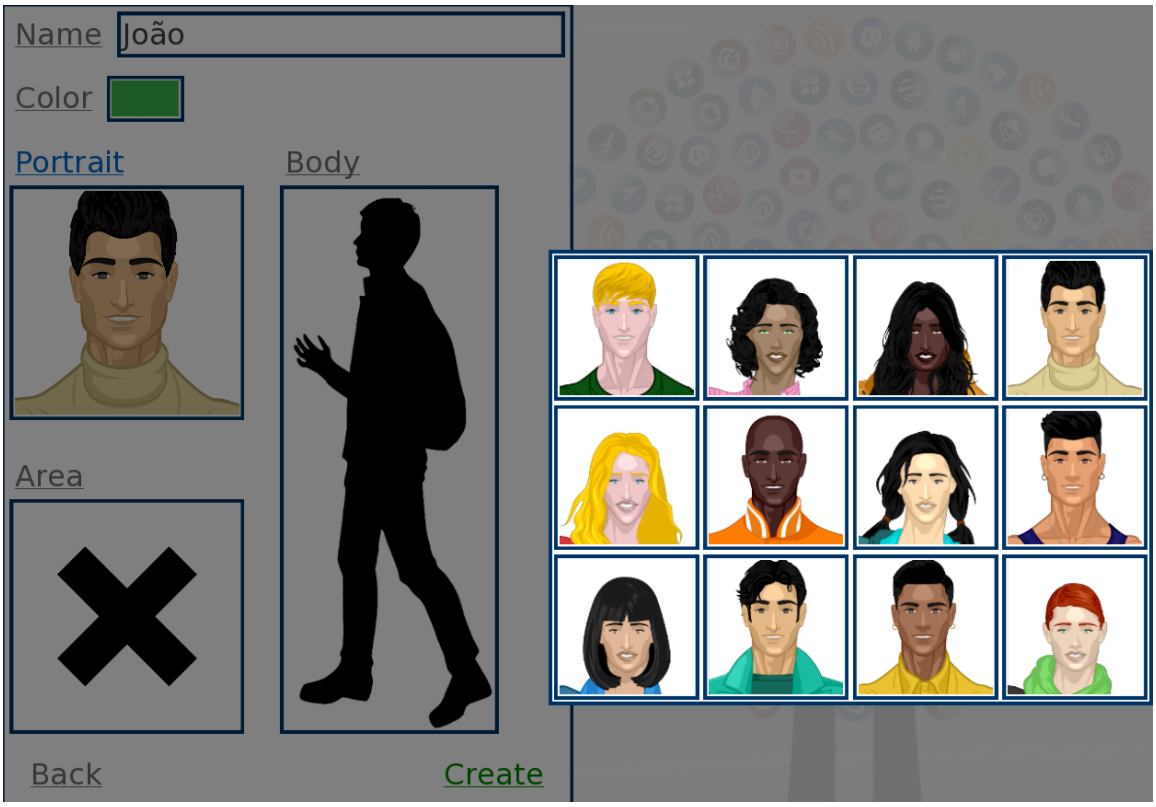


Figure 2.7: Character creation

Figure 2.8: Dialogue creation

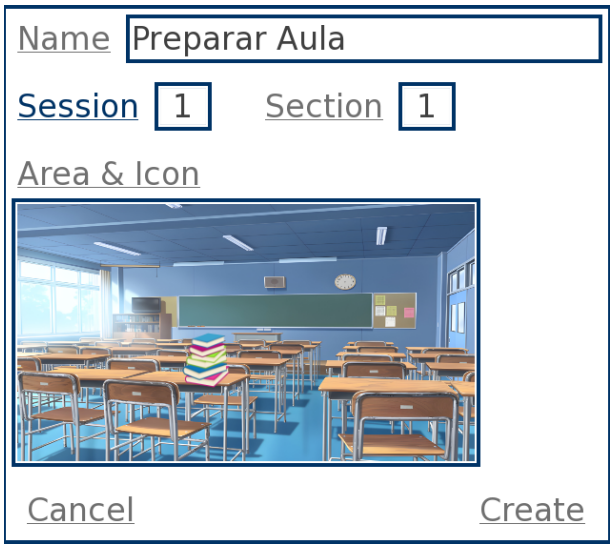


Figure 2.9: Task creation

2.7.2.B Library and Asset Management

The Library section organizes all created content into categorized collections, providing easy access to areas and their connections (figure 2.10), characters, dialogues (figure 2.11), tasks and events. Each content type can be modified or deleted. The system maintains consistency by automatically updating references when content is modified and provides confirmation warnings to prevent accidental deletion of important game elements.

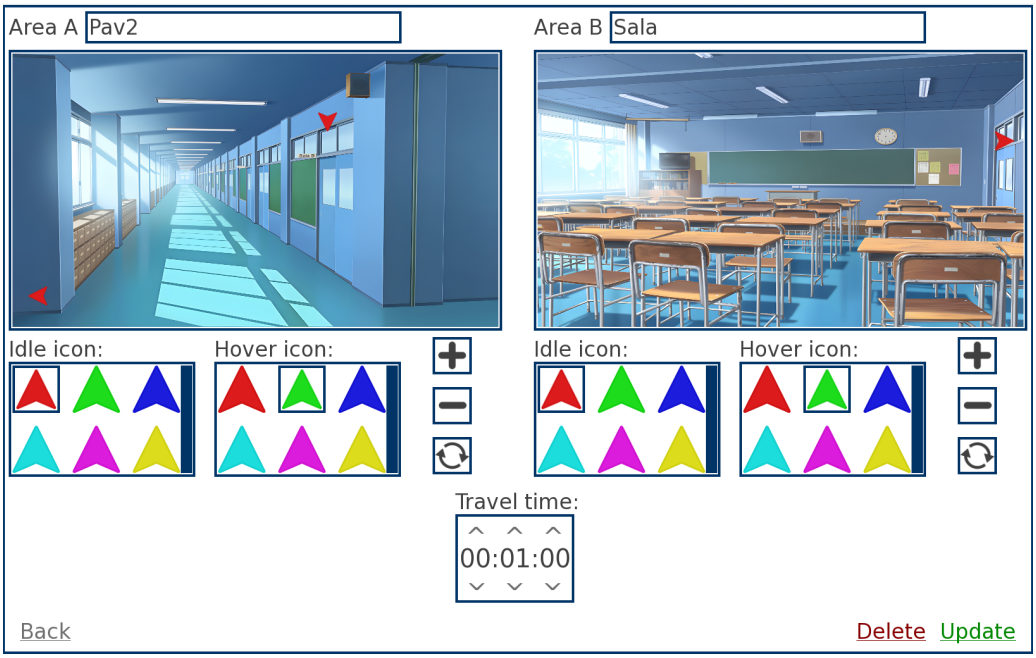


Figure 2.10: Connection edition in Library

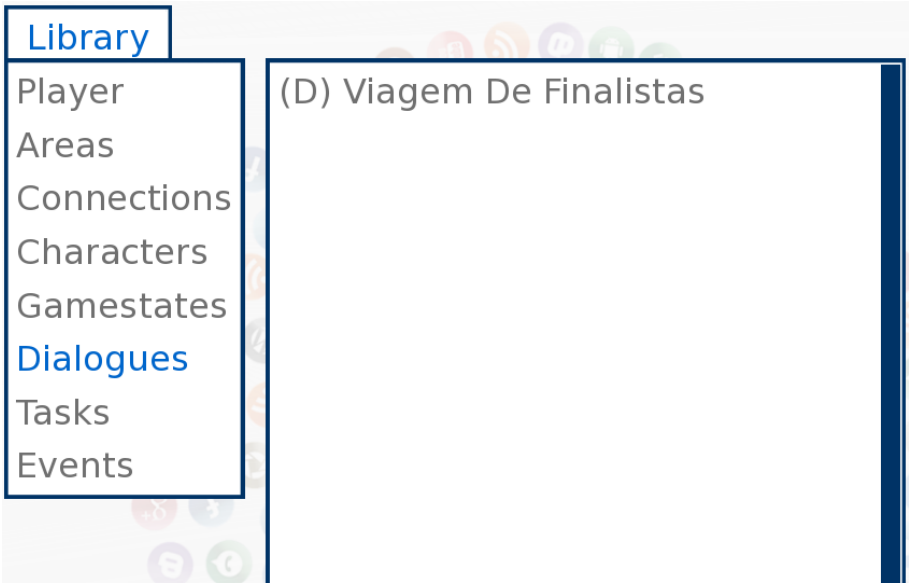
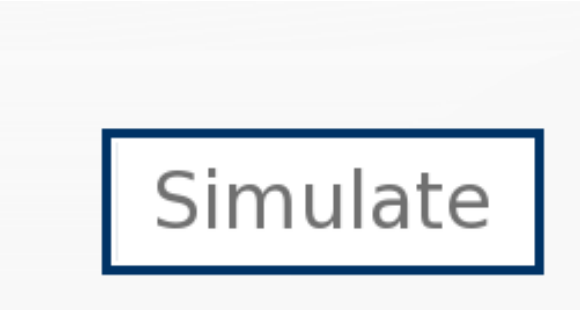


Figure 2.11: Example of dialogue library storage

2.7.2.C Simulator

The editor includes an integrated simulation feature that allows designers to test their created content in a playable game environment. By clicking the “Simulate” button (figure 2.12(a)), users can launch a runtime version of their visual novel that incorporates all the characters, areas, dialogues, tasks, and events they have authored. This simulation functionality serves as a crucial validation tool, enabling designers to experience their created content from the player’s perspective and verify that dialogues flow naturally, character interactions feel authentic, and the overall narrative structure meets their creative expectations within the actual game context (figure 2.12(b)).



((a)) Simulate Button in the main menu

((b)) Simulation example, similar to Brave game

Figure 2.12: Simulation functionality showing the activation button and gameplay interface

2.7.3 Implications for This Thesis

The examination of the existing editor reveals several key limitations that directly inform the contributions of this thesis. While the baseline editor provides fundamental visual novel creation capabilities, it lacks the sophisticated character modeling and AI-assisted dialogue generation features essential for creating consistent and contextually appropriate NPC interactions.

2.7.3.A Identified Limitations

The existing editor's character development system allows only basic property definition without psychological depth or personality modeling. The dialogue creation process relies entirely on manual authoring, requiring designers to craft every line individually without assistance or consistency checking. Additionally, the system lacks contextual awareness mechanisms that could ensure dialogue appropriateness for different social situations, particularly sensitive scenarios involving bullying or cyberbullying dynamics.

2.7.3.B Thesis Contributions

To address these limitations, this thesis introduces several key enhancements that build upon the established editor foundation:

Personality-Driven Character System: The implementation extends the basic character creation with comprehensive personality modeling based on the FFM, enabling characters to exhibit consistent psychological traits across different dialogue situations. This addresses the gap identified in the related work 2.1 regarding the need for psychologically grounded NPC behavior.

Relationship Dynamics: The enhanced system incorporates character relationship modeling (friendships, romantic connections, conflicts) that influences dialogue generation, reflecting the social dynamics research discussed in the personality-driven NPC design section 2.1.

AI-Assisted Dialogue Generation: The integration of LLM technology provides contextual dialogue suggestions, directly addressing the challenges identified in the LLM narrative generation literature regarding consistency and contextual appropriateness. This addresses the limitations discussed in the game development tools and AI integration section 2.6 regarding the need for AI-assisted content creation while maintaining editorial control.

Contextual Parameter System: The implementation includes advanced parameters such as "speaking of" (character targeting), scenario/topic specification, bullying level classification, and emotional state selection that ensure generated dialogues are appropriate for the intended social dynamics and educational objectives. These parameters build upon the dialogue systems research discussed in section 2.4, extending basic emotional expression selection with sophisticated contextual controls.

3

Implementation

Contents

3.1 Overview

This chapter presents the proposed solution for integrating LLMs into a game editor to assist designers in generating NPC dialogues. The system prioritizes a human-in-the-loop workflow in which LLM outputs act as editable drafts rather than final runtime text. By placing the generative step in the development pipeline, the solution aims to reduce latency and cost constraints, preserve editorial control, and enable designers to shape tone, personality, and narrative intent before export.

At a high level, the architecture features an editor interface that allows designers to define characters, their personalities, and relationships with other characters; a prompting and control module that builds structured prompts from those definitions; an integration layer that handles requests to one or more LLM services; and a persistence and filtering layer responsible for ranking variants and exporting final lines to the target engine (Ren'Py). Figure 3.1 provides a schematic overview of these components and their main data flows. The design emphasizes modularity so that components such as the LLM provider or the exporter can be replaced or upgraded independently.

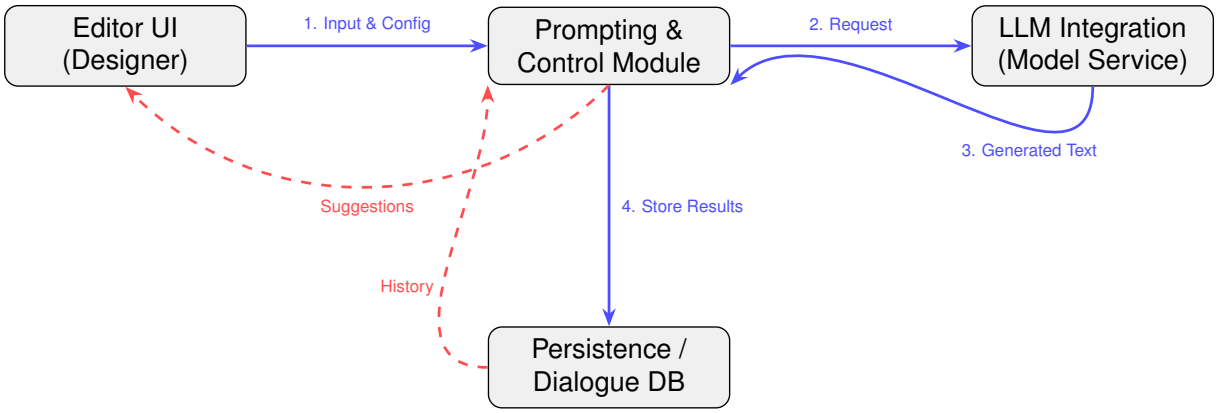


Figure 3.1: System architecture showing the main workflow from designer input to exported dialogue.

The decision to focus on offline generation is deliberate. Runtime generation can enable richer emergent behavior, but it introduces higher operational costs, latency constraints and greater risk of unvetted content appearing during gameplay. By contrast, the design workflow allows teams to use LLMs for large scale drafting while maintaining strict editorial review, improving safety and alignment with design goals.

The key objectives of the proposed system are:

- **trait-aligned generation**: produce dialogue that reflects specified personality parameters and role relationships;
- **contextual relevance**: ensure outputs reflect the current scene information and the recent dialogue history, so that lines suit the situation;

- **workflow efficiency**: save time by creating lots of dialogue quickly, make it easy to edit and retry;
- **traceability**: log prompts, model parameters, generated outputs and any filtering or review decisions so that generation steps can be audited, reproduced and iteratively improved;

Figure 3.2 illustrates the complete workflow that supports these objectives.

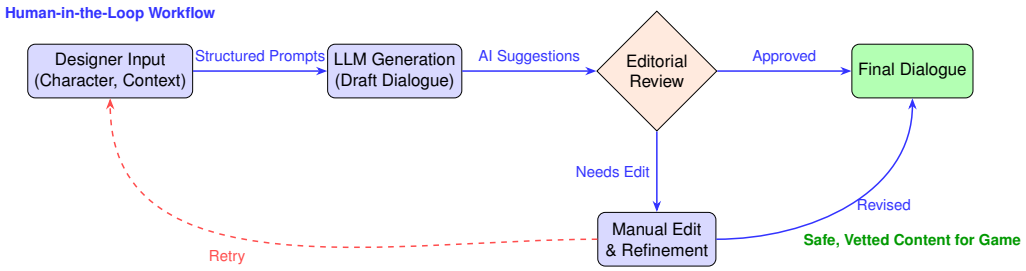


Figure 3.2: Design workflow emphasizing human editorial control and iterative refinement.

The remainder of this chapter details the functional and non-functional requirements, the architecture and each core component. It also presents concrete prompt templates, data formats and an example walkthrough that illustrate how designers will use the system in practice. The evaluation chapter that follows uses the artifacts defined here to measure trait alignment, perceived realism, coherence and user satisfaction in controlled tests.

3.2 Requirements

The proposed solution must address both functional and non-functional requirements in order to be usable by game designers and effective in the generation of dialogues for NPCs. Functional requirements capture what the system should do, while non-functional requirements constrain how it should operate in practice. Additionally, certain restrictions apply due to the technological choices and integration context.

3.2.1 Functional Requirements

The following table summarises the key functional requirements of the system:

Table 3.1: Functional requirements

ID	Description	Priority	Acceptance Criterion
F1	Define NPC personalities using a structured model (e.g., FFM traits and facets).	Must	Designer can create a character profile with adjustable trait values.
F2	Define relationships between characters.	Must	System stores character relationships and uses them as context in dialogue generation.
F3	Configure dialogue parameters: emotion, target of speech, bullying level, topic.	Must	Editor provides controls to select these parameters and they are reflected in prompts.
F4	Generate multiple alternative dialogue lines for the same situation.	Must	For a given prompt, at least N variants are produced.
F5	Allow manual editing of generated dialogues and save different versions.	Must	Edited dialogues persist across sessions and versions can be retrieved.
F6	Support scene context, including participants and setting metadata.	Must	Generated lines reflect the scene metadata provided by the designer.

3.2.2 Non-functional Requirements

The non-functional requirements ensure efficiency, robustness and usability of the solution:

Table 3.2: Non-functional requirements

ID	Description	Priority	Acceptance Criterion
NF1	Dialogue generation time must remain acceptable.	Must	Typical requests complete in under 15 seconds during testing.
NF2	Maintain logs of prompts, outputs and user edits to support traceability.	Must	Log entries exist for all generated dialogues with timestamps.
NF3	Provide a user-friendly interface that requires minimal training.	Should	New designers can generate and edit dialogues after a short tutorial.
NF4	Ensure data privacy for test scenarios and user-provided content.	Must	No personal identifiers are stored in raw logs or exports.

3.2.3 Restrictions

Certain restrictions influence the design and scope of the solution:

- The tool must integrate with the Ren'Py engine, requiring exports to the `.rpy` format.
- The system relies on external LLMs, which impose limits on context window size and token usage.
- Internet connectivity is required for Application Program Interface (API)-based model access.
- Local deployment is restricted by hardware and cost considerations.

3.3 System Architecture

This section presents the detailed architecture of the dialogue generation system, building upon the high-level overview presented in figure 3.1. The architecture follows modular design principles to ensure maintainability, extensibility, and clear separation of concerns.

3.3.1 Architectural Principles

The system architecture is guided by several key principles:

- **Modular Design**: Each component has well-defined responsibilities and interfaces, allowing independent development and testing.
- **Separation of Concerns**: User interface, dialogue generation logic, data persistence, and external integrations are clearly separated.
- **Human-in-the-Loop**: The architecture prioritizes designer control and editorial oversight at every stage.

3.3.2 Component Overview

The system consists of four primary components, each responsible for specific aspects of the dialogue generation workflow:

3.3.2.A Editor Interface

The frontend component provides the primary user interface for game designers through two important editors: the Character Editor and the Dialogue Editor.

A – Character Editor (figure 3.3) manages comprehensive character definition, including personality profiles based on the FFM with five adjustable parameters: Extraversion, Agreeableness, Conscientiousness, Neuroticism and Openness to Experience. Each trait can be set to three values (low, neutral, high) (figure 3.4), generating complex personality facets through trait combinations. The editor implements an advanced personality summary system that converts trait combinations into natural language descriptions using over 100 possible different descriptions A –.1, providing rich psychological portraits for LLM context. Additionally, the Character Editor manages bidirectional relationships between characters (friend, boyfriend, dislike) (figure 3.6), automatically applying mutual relationship updates and maintaining social network consistency.

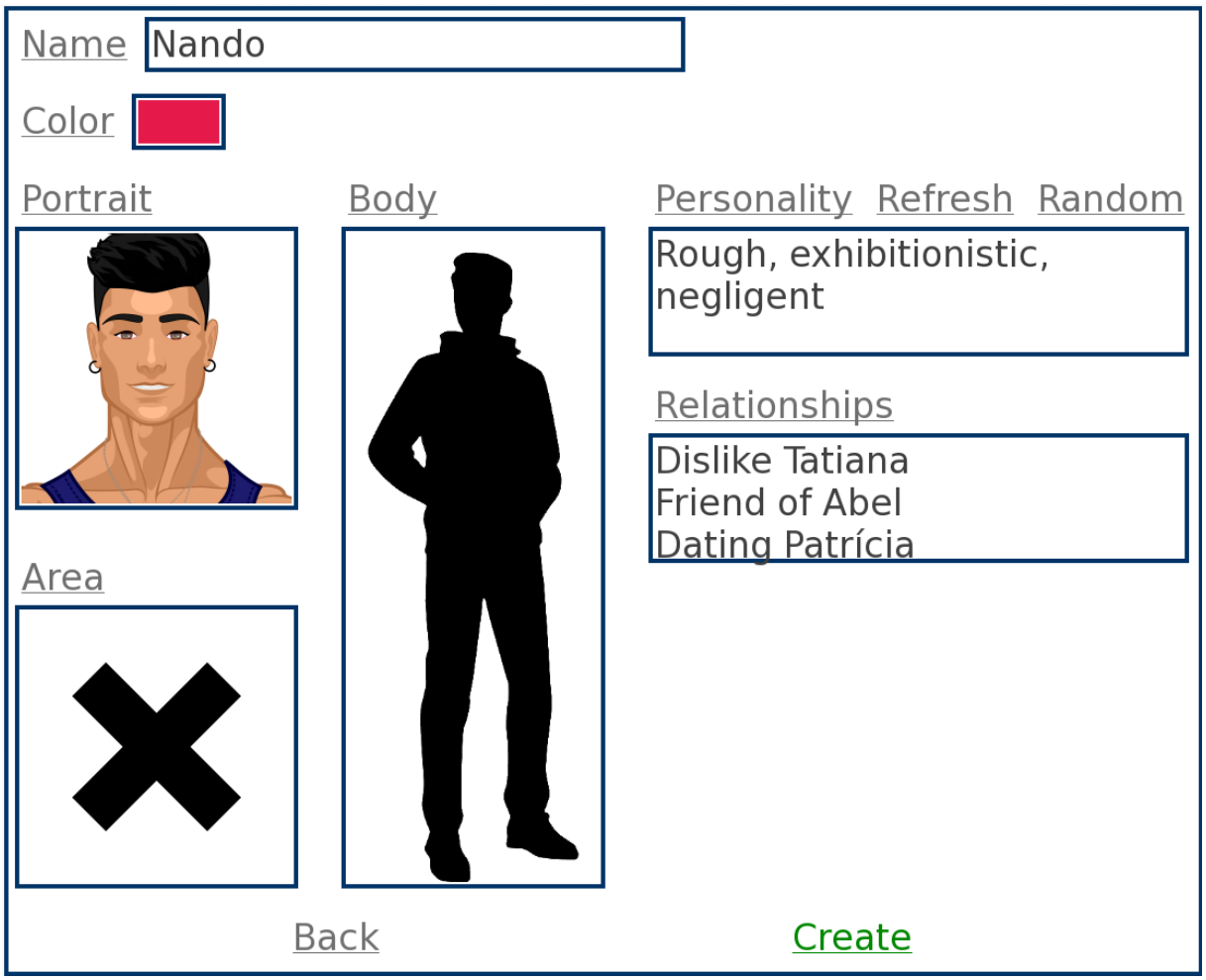


Figure 3.3: Character Editor interface with personality description and relationships example.

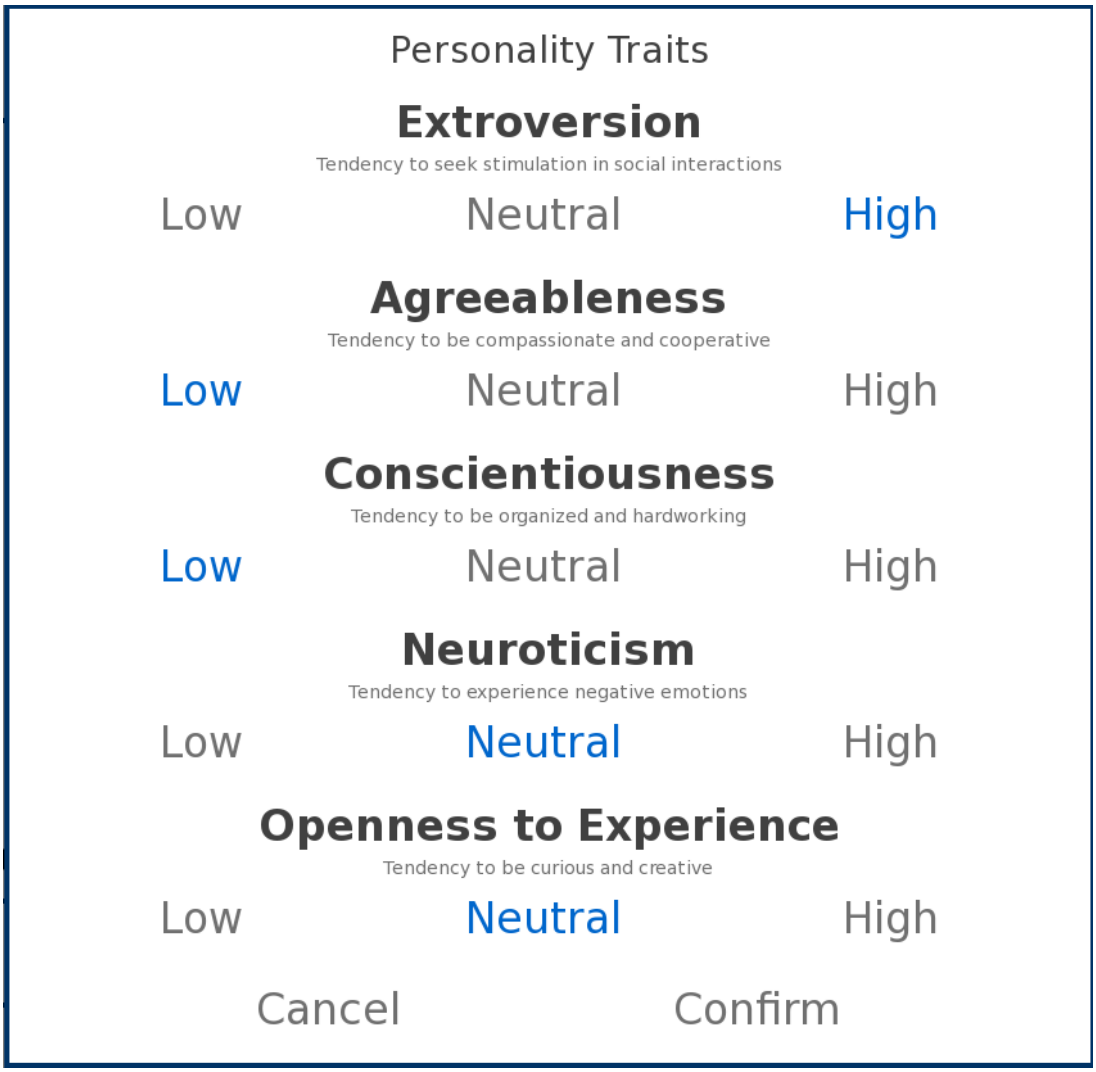


Figure 3.4: Personality configuration with the FFM

A –.1 Facets and FFM trait combinations The system uses a comprehensive facet mapping to generate personality combinations [50, 51]. As shown in figure 3.5, intersections between high trait levels (top row) and low trait levels (left column) produce specific personality facets. The algorithm collects all applicable adjectives from different trait combinations and then randomly selects up to 5 descriptors to avoid overwhelming the LLM with excessive personality information that could lead to confusion in dialogue generation. To provide personality variation while maintaining psychological consistency, the system implements a refresh functionality that allows designers to regenerate personality descriptions by re-running the selection algorithm with the same base trait parameters, producing different combinations of descriptors that remain faithful to the underlying FFM configuration. The pseudo code for this algorithm is shown in A.1.

	Extraversion (I+)	Agreeableness (II+)	Conscientiousness (III+)	Emotional Stability (IV+)	Openness to Experience (V+)
Introversion (I−)		Timid, Unaggressive, Submissive	Reserved, Restrained, Serious	Tranquil, Sedate, Placid	Inner-Directed
Disagreeableness (II−)	Rough, Abrupt, Crude		Rigid, Hard	Insensitive, Unaffectionate, Passionless	Shrewd, Sharp-Witted
Low Conscientiousness (III−)	Reckless, Unruly, Devil-May-Care	Permissive, Enabling		Complacent, Unbothered	Unconventional, Slapdash
Neuroticism (IV−)	High-Strung	Emotional, Gullible	Particular, Intrusive		Paranoid, Histrionic, Weird
Closed to Experience (V−)	Unscrupulous, Pompous	Simple, Dependent, Servile	Muleheaded, Obstinate, Infuriating	Unreflective, Unsophisticated, Imperceptive	

Figure 3.5: Facet combinations table showing intersections between personality trait levels

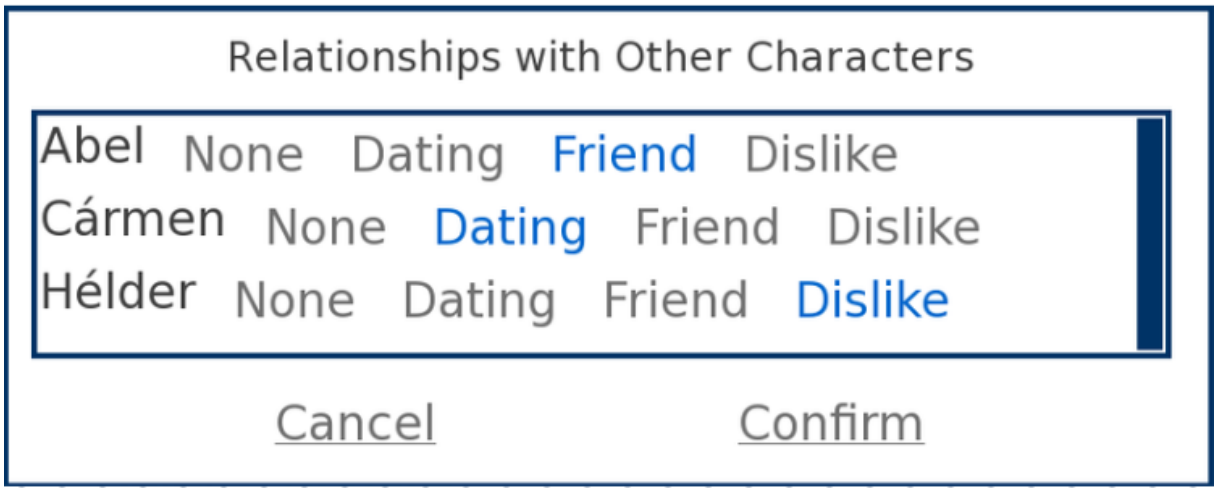


Figure 3.6: Relationships configuration

B – Dialogue Editor The Dialogue Editor enables designers to configure comprehensive dialogue-specific parameters for LLM generation. Key parameters include: speaker selection with emotional state (seven emotions: neutral, anger, sadness, fear, happiness, disappointment, surprise), referenced characters for dialogue context, bullying level classification (None, Low, Medium, High), and scenario/topic (figure 3.8) specification for contextual generation. To enhance usability, emotion portrait images display a tooltip with the emotion name (e.g., “Happiness”) when users hover over them, providing clearer

identification of available emotions (figure 3.7). The editor supports dialogue suggestion through the LLM integration, comprehensive prompt construction using character personalities, relationships and all parameters above, and threaded background processing with timeout handling, to maintain User Interface (UI) responsiveness during API calls that typically take 5-15 seconds, with a loading screen (figure 3.9). If the API request exceeds the 15-second timeout threshold, the system gracefully handles the failure and notifies the user. The detailed implementation for the dialogue editor provided in Appendix A.2.

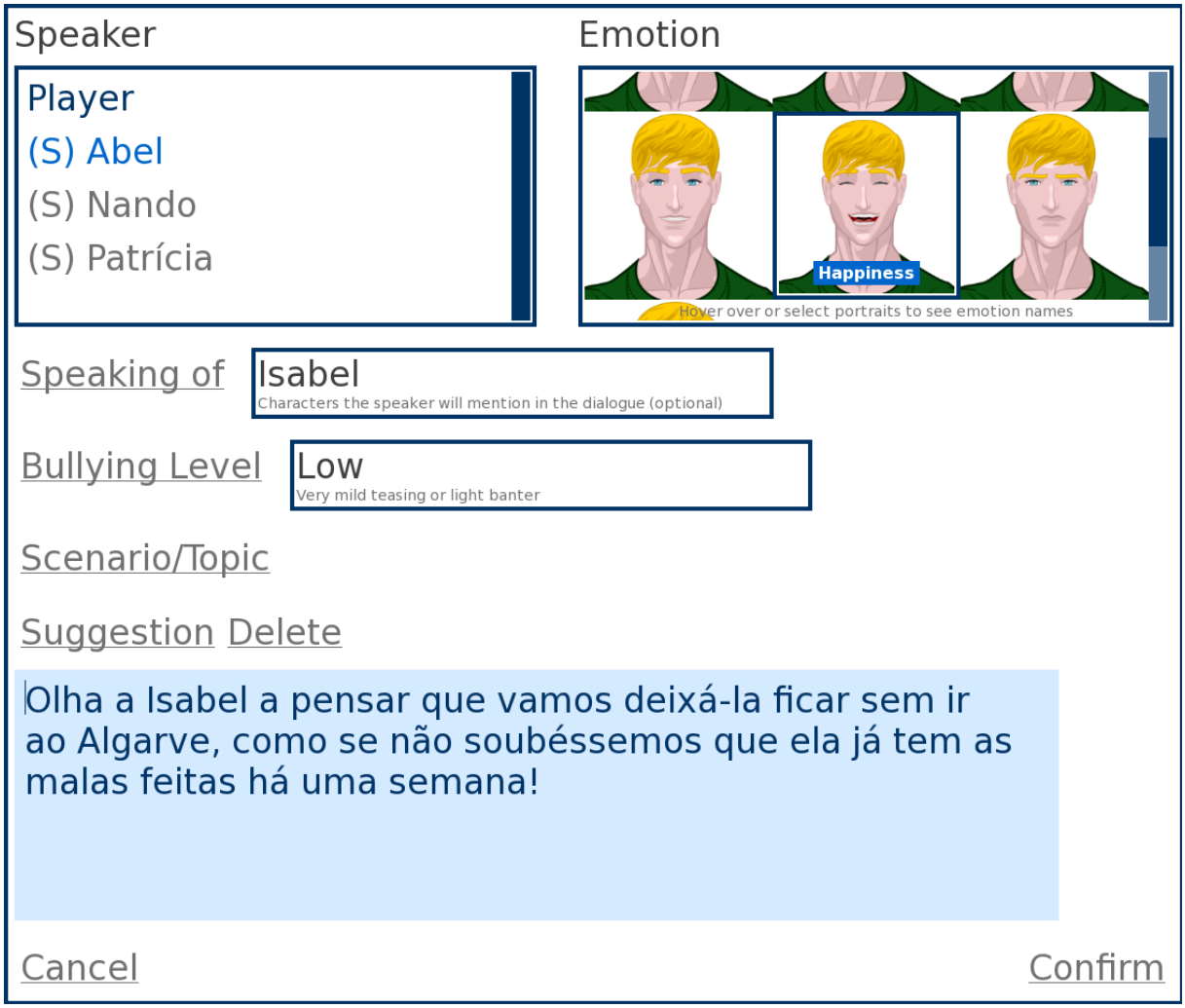


Figure 3.7: Dialogue Editor interface displaying parameter configuration and LLM suggestion workflow.

Figure 3.8: Topic/Scenario example for the LLM

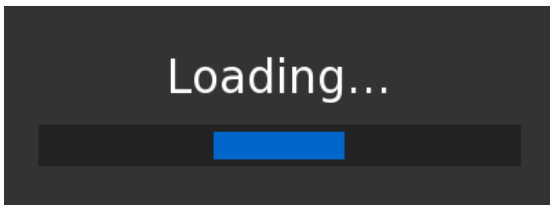


Figure 3.9: Loading screen to give feedback to the user while waiting

C – Integration Workflow Figure 3.10 illustrates how both editors work together in the dialogue generation process.

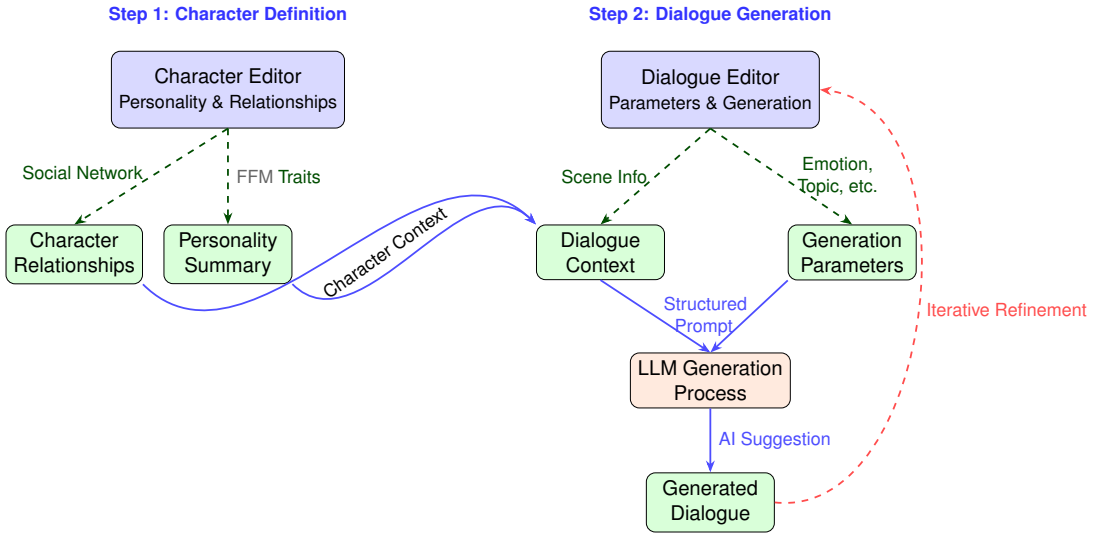


Figure 3.10: Editor interface workflow showing relationship between character definition and dialogue generation.

The workflow demonstrates the sequential dependency between the two editors: character data flows from the Character Editor into the Dialogue Editor's context, where it combines with scene-specific parameters to create structured prompts for the LLM generation process.

3.3.2.B Prompting & Control Module

This component is responsible for building and managing the prompts that are sent to the LLM. In the proposed implementation the system uses a predefined prompt template which is populated dynamically with the metadata provided by the designer (character persona, relationships, scene information and per-line parameters such as emotion, target of speech, bullying level and topic).

The module performs the following tasks:

- fill template placeholders with the current character profile, scene context and dialogue parameters chosen by the designer (figure 3.11);
- send the resulting prompt to the LLM integration layer and receive the generated dialogue line suggestion;
- record the prompt and the model output for traceability and versioning.

```
elif speaker_id == 0:
    speaker_text = "A generic NPC"
else:
    speaker_text = gameworld.chars[speaker_id].name
    prompt += f"Speaker name: {speaker_text}\n"

# Add emotion information with clearer emphasis
if emotion_id != -1:
    prompt += f"Speaker's current emotion: {get_emotion_name(emotion_id)}\n"
```

Figure 3.11: Template placeholders code example for character name and emotion

If the designer does not modify any parameters, the module uses a default generic prompt. Different combinations of user-selected parameters therefore produce different filled prompts, while the underlying template logic remains constant. Hyper parameters of the model (for example `temperature` or `max_tokens`) are not changed by this module and remain fixed according to the deployment configuration. A detailed example of a complete prompt template used for dialogue generation between characters is provided in figure 3.12.

This design keeps prompt construction deterministic and simple to audit, while allowing the editor to express a wide range of behaviors through parameter combinations. Prompt versioning and logging enable reproducibility and help with iterative prompt improvements during the evaluation phase.

```
Generate a single line of dialogue, IN EUROPEAN PORTUGUESE, for a character in a Visual
Novel game, following these detailed specifications:

=== 1. SPEAKER INFORMATION ===
Speaker name: Nando
Speaker's current emotion: Happiness
Speaker's personality traits: Explosive, firm, strict, temperamental and particular
The dialogue should reflect these personality traits.

=== 2. SETTING AND CONTEXT ===
Game setting: High school environment with 17-year-old student characters
Conversation topic/scenario: Praia
The dialogue MUST address this specific topic or scenario.
Characters present in the conversation: Tatiana; Isabel
The speaker is aware these characters are listening.

=== 3. CHARACTERS BEING MENTIONED ===
The speaker (Nando) will talk about: Isabel
Details about mentioned characters:
- Isabel's personality: Dramatic, permissive, idealistic and compulsive

=== 4. SOCIAL RELATIONSHIPS ===
Speaker's relationships with others:
- Nando and Abel are friends
- Nando and Tatiana dislike each other

=== 5. TONE AND STYLE ===
Tone: Include very mild teasing that could be interpreted as either playful
or slightly mean, the dialogue should have light teasing that treads the
line between friendly banter and mild mockery.

=== 6. CONVERSATION HISTORY ===
Previous lines in this conversation:
Isabel(Happiness): "Oh Nando, se fores para a praia com essa cara, o mar vai recuar de medo!"
Next line to generate: Nando (Happiness): [GENERATE THIS LINE]
The dialogue should naturally follow from this conversation history.

=== 7. OUTPUT FORMAT REQUIREMENTS ===
1. Write exactly ONE line of dialogue in European Portuguese
2. Make the dialogue sound natural for a Portuguese teenager in a school setting
3. Do NOT include quotation marks, asterisks, or character names
4. Do NOT include narration, descriptions, or any text that isn't spoken dialogue
5. Return ONLY the dialogue text itself, nothing else
6. The dialogue MUST be relevant to the specified topic/scenario
7. The dialogue should clearly convey the speaker's happy emotion
```

Figure 3.12: Example of a complete prompt template used for dialogue generation. One of the LLM suggestions for this prompt was: *"Muito engraçada... Vê lá se a maré não começa a encher com o teu choro!"*.

3.3.2.C LLM Integration Layer

The integration layer is specialized for a single model, `deepseek-chat-v3-0324:free`, accessed via the OpenRouter gateway (`openrouter.ai`). This component is responsible for sending requests to the provider, parsing model responses, and basic error handling with timeout management and logging. In our tests the chosen provider has been reliable; nevertheless the layer includes a configurable local fallback option using Ollama with a smaller local model to increase resilience or support offline operation.

All requests and responses are recorded for traceability. Log records exclude or anonymize any sensitive user data according to the project's privacy policy, and the integration layer exposes configuration points for switching providers, adjusting timeout policies and enabling the local fallback when required. The detailed implementation pseudo code for the LLM integration is provided in Appendix A.4.

3.3.2.D Persistence Layer

The persistence layer leverages Ren'Py's built-in data management system to handle all storage requirements. The implementation uses three main storage mechanisms: (1) `default` variables for session-persistent data like character definitions, dialogue configurations, and editor states that persist during gameplay sessions but reset between application restarts; (2) `persistent` variables for cross-session data such as personality summary cache that maintains generated character descriptions between application sessions; and (3) the central `gameworld` object that stores the complete game configuration including areas, characters, dialogues, tasks, and events.

The system implements a dual-layer architecture with a editor-time `Gameworld` class for editor operations and a runtime `Gameworld_RUNTIME` class that creates deep copies of design data for simulation execution, used for the evaluation tests so users can try a prototype version of the created game. Character relationships are stored as dictionaries within character objects, while dialogue generation requests are cached to avoid redundant API calls. The persistence layer automatically handles Ren'Py's save file management through the configured save directory, ensuring that all game progress and editor configurations are properly maintained across sessions.

3.3.3 Data Flow Architecture

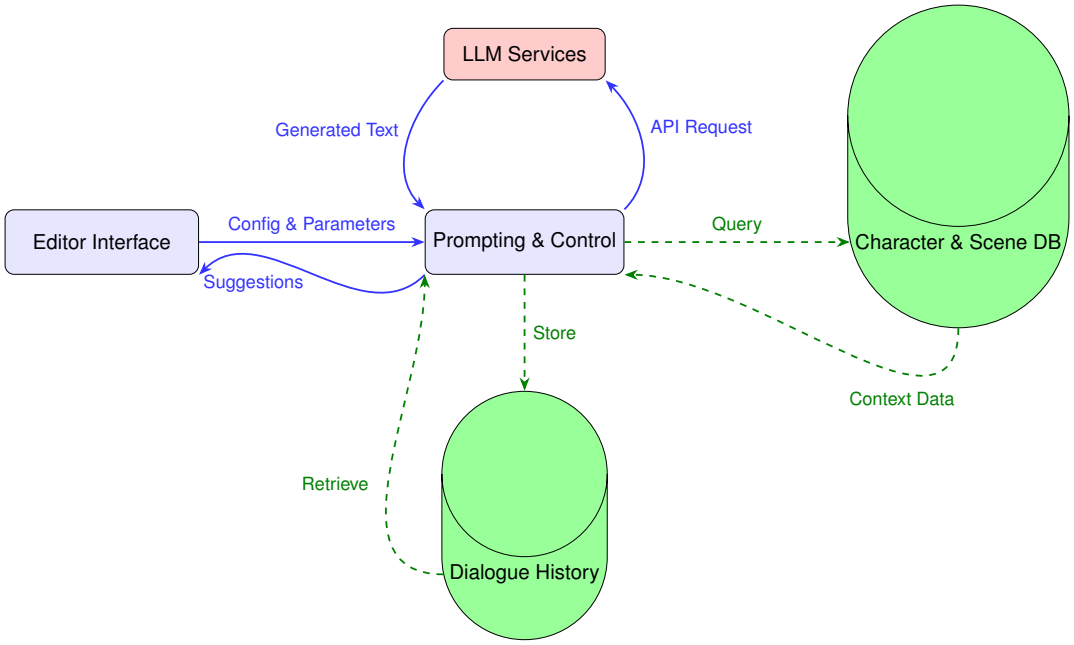


Figure 3.13: Detailed data flow architecture showing component interactions.

3.3.4 Communication Patterns

The architecture employs some communication patterns to ensure efficient and reliable operation:

3.3.4.A Request-Response Pattern

Primary interactions between the Editor Interface and backend components follow a synchronous request-response pattern, providing immediate feedback to designers.

3.3.4.B Asynchronous Background Processing

LLM generation requests are processed in background threads to maintain UI responsiveness during API calls. The system uses Python threading with timeout monitoring and main thread callbacks to update the interface once generation completes, preventing UI freezing during the typical 5-15 second API response times.

3.3.4.C Circuit Breaker Pattern

The LLM Integration Layer implements circuit breaker patterns to handle external service failures gracefully and prevent cascade failures. This approach maintains system stability when facing API rate limits, network issues, or service outages, allowing graceful degradation rather than complete system failure.

3.3.5 Scalability Considerations

While the system is designed for editor-time use rather than high-throughput scenarios, the implemented architecture includes one key design decision that supports reasonable scalability:

- **Asynchronous Processing**: LLM generation requests are processed in background threads to maintain UI responsiveness during API calls, preventing interface blocking during the typical 5-15 second response times from external services.

The current implementation prioritizes simplicity and functionality over advanced scalability features, making it well-suited for individual designers or small teams working on game dialogue generation during development phases.

3.3.6 Privacy and Ethical Considerations

The current implementation includes basic privacy considerations appropriate for a research prototype:

- **Local Data Storage**: Character definitions, dialogue configurations, and generated content are stored locally using Ren'Py's built-in storage mechanisms, ensuring user data remains on the local machine rather than being transmitted to external servers beyond the necessary LLM API calls.
- **Minimal External Data Transmission**: Only the constructed prompts containing character context and dialogue parameters are sent to the LLM service, with no storage of user data on external servers.

As a research prototype focused on demonstrating LLM integration capabilities, the system prioritizes functional implementation over comprehensive privacy infrastructure. Production deployments would require additional privacy-preserving mechanisms such as data anonymization, audit logging, and access control systems based on specific organizational requirements.

A demonstration video showcasing the implemented dialogue generation system is available online at: https://youtu.be/t2H84_mc-V4. For convenience, the video can also be accessed via the QR code below.

Figure 3.14: Demo

4

Experiment

Contents

4.1 Introduction

This chapter presents the experimental design and methodology employed to evaluate the LLM-based dialogue generation system described in Chapter 3. The experimental approach is designed to systematically assess the system's performance and usability for dialogue creation workflows, focusing on content quality, user experience, and satisfaction with AI-assisted creative processes.

The experiment addresses the core research question of whether LLMs can successfully assist game designers in generating contextually appropriate and character-aligned dialogue content. The experimental methodology employs a mixed-methods approach combining quantitative measurements with qualitative insights to provide a comprehensive assessment of system performance. The primary experimental objectives are:

- to design an experimental methodology for evaluating AI-assisted creative tools in game development contexts;
- to establish metrics and measurement frameworks for assessing dialogue generation quality, usability, and user satisfaction;
- to create controlled experimental conditions that reflect realistic game development scenarios;
- to develop data collection procedures that capture both measurable performance indicators and subjective creative experiences.

The experimental protocol involves structured user testing sessions where participants complete predefined dialogue creation tasks using the integrated system, followed by comprehensive assessment through questionnaires and observational data collection. The methodology is designed to provide generalizable insights while acknowledging the naturally subjective character of creative content evaluation.

4.2 Evaluation Methodology

This section describes the methodology employed to evaluate the LLM-based dialogue generation system, including participant recruitment, study design, and data collection procedures.

4.2.1 Mixed-Methods Approach

The evaluation employs a mixed-methods research design that combines quantitative measurements with qualitative insights to provide a comprehensive assessment of the system's performance. This approach enables the capture of both measurable usability metrics and nuanced user experiences that purely quantitative methods might miss. The methodology integrates:

- **Quantitative data collection**: Through structured Likert-scale questionnaires measuring usability, content quality, and user satisfaction
- **Qualitative feedback**: Via open-ended questions and behavioral observations during testing sessions
- **Triangulation**: Combining different types of feedback (questionnaire ratings, written or oral comments, and direct observations) to get a complete picture of system performance

The mixed-methods approach ensures robust validation of results while accommodating the subjective nature of creative content evaluation and user experience assessment.

4.2.2 Ethical Considerations

The study was conducted in accordance with standard research ethics protocols:

- **Informed consent**: All participants provided explicit consent before participation, with clear information about the study's purpose and procedures
- **Voluntary participation**: Participants were informed of their right to withdraw at any time without consequences
- **Data privacy**: All collected data was anonymized and stored securely in compliance with General Data Protection Regulation (GDPR)
- **Confidentiality**: Personal identifiers were separated from response data, with a 5-year data retention policy

4.3 Experimental Design and Testing Procedures

This section describes the structured experimental design employed to evaluate the LLM-based dialogue generation system, including the testing protocol, task design, and data collection procedures.

4.3.1 Study Design Overview

The evaluation employed an observational study design where all participants experienced the complete system workflow. This approach ensures that each participant provides comprehensive feedback on all system components while allowing systematic assessment of user experience with the AI-assisted dialogue generation system.

The study was structured as a single-session evaluation lasting approximately 40 minutes per participant, consisting of:

- **Pre-test phase** (5 minutes): Demographic questionnaire and baseline experience assessment
- **Training phase** (5 minutes): Editor introduction and guided tutorial
- **Task execution phase** (15 minutes): Hands-on system usage following structured scenarios
- **Gameplay phase** (5 minutes): Testing the created content through simulation
- **Post-test phase** (10 minutes): Experience and satisfaction questionnaires

4.3.2 Testing Protocol

Each testing session followed a standardized protocol to ensure consistency across all participants:

4.3.2.A Session Preparation

- After a brief oral introduction, participants received an information sheet explaining the study context and objectives
- The testing environment was prepared with a computer running the final version of the editor ready to be tested
- A comprehensive step-by-step test guide was provided to walk participants through each phase of the evaluation (see in Appendix A.4.1)
- A character relationship diagram and personality reference sheet were provided as supporting materials for the predefined scenario (see in Appendix A.4.2)
- The researcher observed participants throughout the session, taking notes on user behavior and interactions

4.3.2.B Scenario Context

To provide a realistic and engaging context for evaluation, a comprehensive scenario was developed set in a Portuguese high school environment. The scenario involves a class preparing for their final school trip to the Algarve, with complex social dynamics among the students:

Main Characters and Relationships:

- **Estrela**: Class representative, responsible and empathetic, organizing the trip
- **Cármen**: Cannot attend the trip due to parental restrictions, accuses Estrela of favoritism toward her friend Isabel
- **Tatiana**: Shy and passive, victim of bullying by Patrícia, Manuela, and Cármen

- **Patrícia**: Part of the bullying group, has a boyfriend (**Nando**) who wants to defend her
- **Rui**: Supports Tatiana against injustice, opposes the bullying behavior
- **Isabel**: Close friend of Estrela, enthusiastic about the Algarve trip

Conflict Dynamics: The scenario includes multiple layers of conflict: Tatiana previously created a fake profile to deceive Patrícia and sent offensive messages as revenge for the bullying. This action created additional tension, with Nando (Patrícia's boyfriend) wanting to defend her, while Rui and Estrela support Tatiana against the ongoing injustice.

This rich social context provides numerous opportunities for testing dialogue generation across different emotional states, relationship types, and conflict situations.

4.3.2.C Task Structure

The evaluation centered around a predefined scenario set in a high school environment, involving character relationships, conflicts, and social dynamics explained before. Most of this scenario is already defined in the editor but there are some incomplete characters to edit and all dialogues to create. Participants were required to complete three main tasks:

Task 1: Character Editing Participants edited three incomplete characters (Estrela, Patrícia, and Rui) by:

- Defining personality traits using the FFM interface
- Establishing social relationships and connections with other characters
- Adjusting character attributes to match the provided scenario context

Task 2: Dialogue Creation Participants created three distinct dialogues with specific objectives using the AI assistance:

- **Dialogue 1**: Characters - Tatiana, Isabel, and Estrela; Topic - School trip planning; Objective - Demonstrate friendship dynamics and personality traits
- **Dialogue 2**: Characters - Abel, Jorge, Samuel, and Rui; Topic - Free choice; Objective - Test natural interactions based on established relationships
- **Dialogue 3**: Characters - Cármen, Manuela, Patrícia, and Nando; Topic - Discussion about Tatiana; Objective - Illustrate social exclusion and group dynamics

Task 3: System Testing Participants experienced their created content through:

- Gameplay simulation using the integrated game engine

- Navigation through the virtual school environment
- Observation of their dialogues in the intended interactive context

4.3.3 Data Collection Methods

Multiple data collection methods were employed to capture comprehensive evaluation data:

4.3.3.A Quantitative Metrics

- **Likert-scale questionnaires**: 5-point scales measuring usability, content quality, and satisfaction
- **Usage analytics**: Number of AI suggestions requested per line and AI suggestion editing frequency
- **Performance metrics**: Success rates for task completion and system error occurrences

4.3.3.B Qualitative Data

- **Open-ended survey responses**: Written feedback on system strengths, weaknesses, and improvement suggestions
- **Behavioral observations**: The researcher took notes on user interaction patterns, hesitations, and problem-solving approaches
- **Verbal comments**: Spontaneous participant feedback during task execution and a brief conversation at the end, recorded in observation notes

4.3.4 Scenario Design Rationale

The selected high school scenario offered several strategic advantages for evaluation:

- **Familiarity**: All participants could relate to school social dynamics and relationships
- **Complexity**: The scenario includes multiple character types, conflicts, and emotional states
- **Realistic dialogue opportunities**: Natural conversation contexts for testing AI-generated content
- **Clear evaluation criteria**: Well-defined character motivations enable assessment of personality consistency

The scenario incorporated various social dynamics including friendship, conflict, bullying, and romantic relationships, providing rich contexts for evaluating the system's ability to generate contextually appropriate and character-consistent dialogues.

4.4 Evaluation Metrics and Measurement Approaches

This section defines the specific metrics used to evaluate the LLM-based dialogue generation system and describes the measurement approaches employed to assess system performance across multiple dimensions.

4.4.1 Evaluation Framework

The evaluation framework was designed to assess four key dimensions of the dialogue generation system:

- **Usability and Interface Design**: How effectively users can interact with the system
- **AI-Generated Content Quality**: The quality and appropriateness of AI-suggested dialogues
- **System Parameters and Customization**: The effectiveness of available controls and options
- **Overall User Experience**: Satisfaction with the complete workflow and final results

4.4.2 Usability Metrics

Usability was assessed through structured questionnaires measuring participants' experience with the editor interface:

- **Ease of use**: Participants rated how easy it was to use the editor
- **Character editing comfort**: Comfort level when editing character personalities and relationships
- **Dialogue comfort**: Comfort level when navigating and creating dialogues
- **Interface flexibility**: Perceived adaptability of the interface to user decisions

4.4.3 Content Quality Metrics

The quality of AI-generated dialogues was evaluated through multiple assessment criteria:

4.4.3.A AI Suggestion Usage

- **Adoption rate**: Frequency of AI suggestion usage (measured as percentage of participants using suggestions)
- **Suggestion quantity**: Number of AI suggestions requested per dialogue line created
- **Edit frequency**: Number of modifications made to AI-generated suggestions

4.4.3.B Content Quality Assessment

Participants evaluated AI-generated dialogues across several dimensions using 5-point Likert scales:

- **Personality consistency**: Whether dialogues matched predefined character personalities
- **Contextual coherence**: Appropriateness of dialogues within the given scenario
- **Naturalness and realism**: How lifelike and believable the dialogues appeared
- **Logical sequence**: Whether dialogue lines followed a coherent progression
- **Conflict representation**: Effectiveness in conveying character relationships and conflicts

4.4.4 System Parameter Effectiveness

The evaluation assessed how well the available parameters supported dialogue creation:

4.4.4.A Parameter Sufficiency

- **Parameter adequacy**: Whether available parameters (emotion, topic, bullying level, etc.) were sufficient for guiding dialogue creation
- **Missing parameter identification**: Open-ended feedback on additional parameters that would be useful

4.4.5 Overall Experience Metrics

The final evaluation dimension captured participants' overall satisfaction and experience:

4.4.5.A Simulation and Integration

- **Expectation alignment**: Whether the final simulation met participant expectations
- **Dialogue coherence**: Naturalness of dialogue sequences in the game context
- **Creative reflection**: How well the final result reflected participants' intended dialogues

4.4.5.B Satisfaction and Feedback

- **Overall satisfaction**: General satisfaction with created dialogues
- **System strengths**: Open-ended feedback on most appreciated features
- **Improvement suggestions**: Recommendations for system enhancements

- **Future usage intent**: Likelihood of using similar tools for content creation

4.4.6 Data Analysis Approach

The collected data was analyzed using both quantitative and qualitative methods:

4.4.6.A Quantitative Analysis

- **Descriptive statistics**: The analysis included mean scores and frequency distributions for all Likert-scale responses
- **Usage pattern analysis**: Analysis of AI suggestion usage patterns and editing behaviors
- **Correlation analysis**: Relationships between user experience levels and system usage patterns

4.4.6.B Qualitative Analysis

- **Thematic analysis**: Identification of common themes in open-ended feedback responses
- **Behavioral observation synthesis**: Compilation and analysis of observation notes
- **Improvement categorization**: Systematic categorization of suggested system enhancements

5

Results

Contents

5.1 Introduction

This chapter presents the comprehensive results obtained from the evaluation study of the LLM-based dialogue generation system described in Chapter 4. The findings are organized to provide a systematic analysis of system performance across the evaluation dimensions established in the experimental methodology: usability assessment, AI-generated content quality, system parameter effectiveness, and overall user experience.

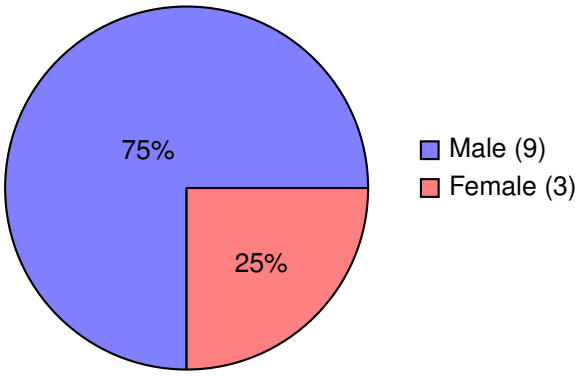
The results are based on data collected from 12 participants who completed the structured evaluation protocol, involving hands-on experience with character definition, dialogue generation workflows, and gameplay simulation. The analysis combines quantitative metrics from Likert-scale questionnaires with qualitative insights from open-ended feedback and behavioral observations, providing a comprehensive assessment of the system's effectiveness in supporting AI-assisted dialogue creation for game development.

5.2 Participant Characteristics and Sample Description

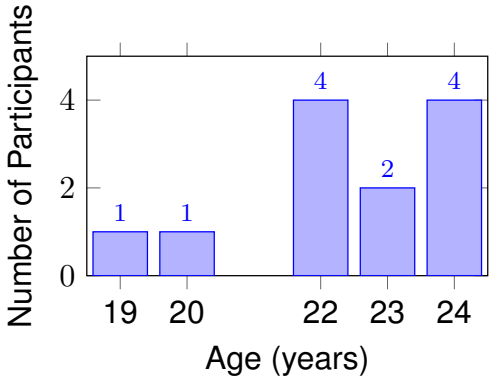
A total of 12 participants were recruited for the evaluation study through convenience sampling from the academic community. The participant demographics provide important context for interpreting the results:

5.2.1 Demographic Characteristics

- **Gender distribution**: 9 male participants (75%) and 3 female participants (25%)
- **Age range**: 19-24 years, with the majority (9 participants) aged 22-24 years
- **Educational background**: Primarily Computer Engineering students (7 out of 12 participants)
- **Native language**: All participants (100%) were native Portuguese speakers
- **Study language**: All evaluation sessions were conducted in Portuguese, including instructions, questionnaires, and participant interactions
- **Technical expertise**: Mixed experience levels with game development tools and programming, including a non-technical participant from Psychology



((a)) Gender Distribution



((b)) Age Distribution

Figure 5.1: Participant demographic characteristics showing gender and age distributions.

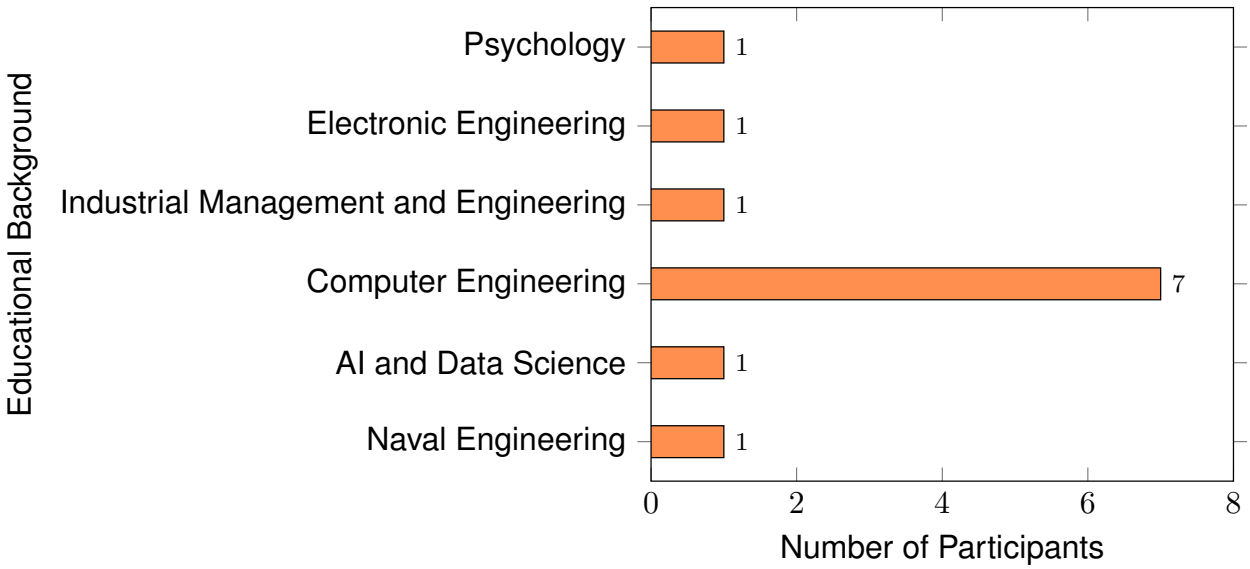


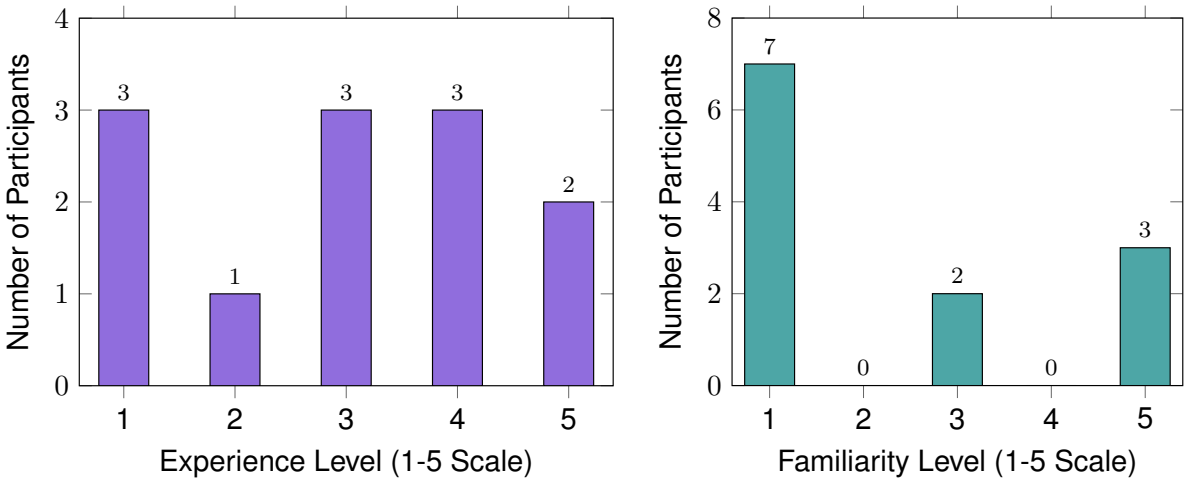
Figure 5.2: Educational background distribution of study participants.

5.2.2 Baseline Experience Assessment

Pre-test questionnaires revealed varied experience levels among participants:

- **Visual Novel familiarity**: 7 out of 12 participants had no prior experience with visual novels
- **Game development experience**: Participants self-rated their experience from 1 (beginner) to 5 (expert), with an average rating of 3.0 indicating moderate experience levels
- **AI tool usage**: High interest in AI-assisted content creation tools, with 9 out of 12 participants expressing interest or strong interest

Notably, the participant sample achieved a remarkably uniform distribution across both game development experience and visual novel familiarity dimensions, as shown in figure 5.3. This uniform distribution is particularly advantageous for evaluation validity, as it ensures that the findings represent diverse user viewpoints rather than being skewed toward any particular experience level, thereby strengthening the generalizability of the results across different user expertise ranges.



((a)) Game Development Experience

((b)) Visual Novel Familiarity

Figure 5.3: Baseline experience assessment showing participants' self-rated experience levels.

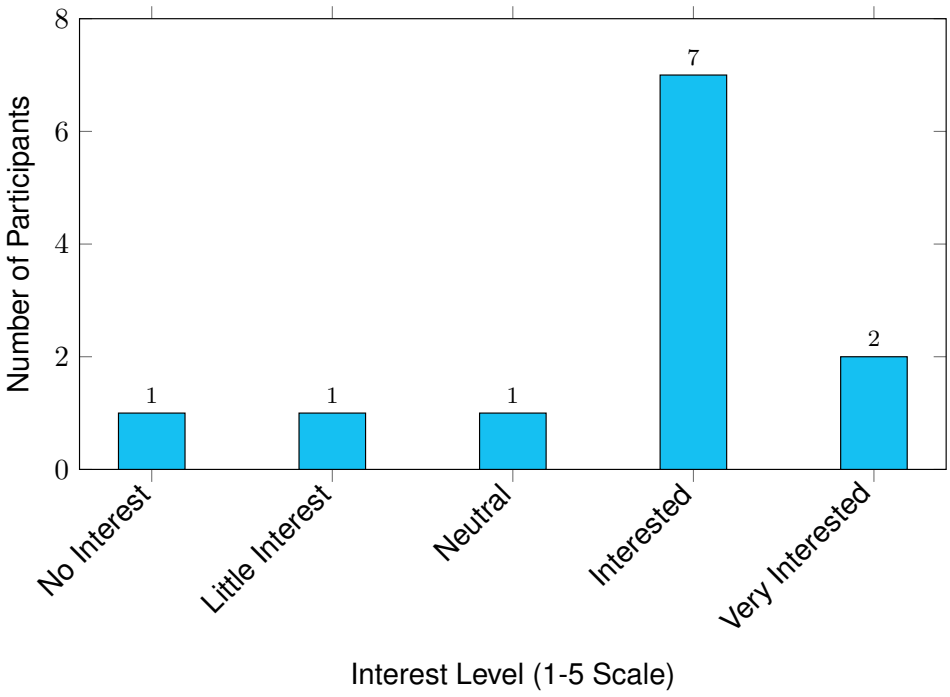


Figure 5.4: Interest in AI-assisted content creation tools among participants.

Notably, while the 3 participants with low interest in AI-assisted tools (25% of the sample) could potentially exhibit negative biases toward AI-generated dialogues or AI assistance in general, this concern did not materialize in the evaluation results. These participants demonstrated similar adoption patterns and provided content quality ratings consistent with the overall positive trends, suggesting that system utility and effectiveness transcended initial AI skepticism.

This demographic profile represents the target audience of technically-oriented individuals who might engage with game development tools, while the mixed experience levels provide diverse perspectives on system usability and effectiveness.

5.2.3 Psychology Expert Perspective

The inclusion of a Psychology expert among the participants provided a preliminary professional perspective on the psychological validity of the character modeling approach. While this represents only a single expert opinion and cannot be considered definitive validation, this participant's expertise in psychological assessment and personality theory offered valuable insights into the effectiveness of the FFM integration within the dialogue generation system. Her feedback indicated that the psychological accuracy of character personality representations was appropriate and that the AI-generated dialogues demonstrated appropriate personality-consistent behaviors and speech patterns. Importantly, her assessment revealed no obvious contradictions or fundamental flaws that would undermine the theoretical foundation of the character modeling system. While this single expert perspective cannot establish comprehensive psychological validity, it provides an encouraging indication that the technical implementation does not contradict established psychological principles and suggests the approach may successfully bridge computational creativity with psychological theory.

5.3 Results and Analysis

This section presents the quantitative and qualitative results obtained from the evaluation study conducted according to the methodology described in Chapter 4, providing a comprehensive analysis of system performance across the defined evaluation dimensions, the questionnaire administered after the evaluation session is shown in A.4.3.

5.3.1 Participant Completion and Engagement

All 12 participants successfully completed the evaluation protocol, achieving a 100% completion rate. The structured 40-minute sessions proved effective in capturing comprehensive feedback while maintaining participant engagement throughout the evaluation process.

5.3.2 Usability and Interface Assessment

The usability evaluation revealed generally positive reception of the editor interface, despite being the most criticized aspect of the system, with participants demonstrating varying levels of comfort across different system components.

5.3.2.A Interface Usability Results

Figures 5.5, 5.7, and 5.6 present the distribution of responses across key usability metrics, classified by game development experience level. The results show:

- **Editor ease of use** (figure 5.5): 41.67% of participants (5/12) found the interface neutral to use, while 41.67% (5/12) expressed positive agreement (ratings 4-5), and 16.66% (2/12) experiencing difficulties
- **Character editing comfort** (figure 5.6): 75% of participants (9/12) expressed comfort with personality and relationship editing features, demonstrating effective implementation of the FFM personality interface
- **Dialogue creation comfort** (figure 5.7): 66.67% of participants (8/12) felt comfortable navigating and creating dialogues (ratings 4-5), indicating a better interface design for the core functionality
- **Interface flexibility**: 66.67% of participants (8/12) perceived the interface as adaptable to their decisions (ratings 4-5), suggesting good workflow accommodation

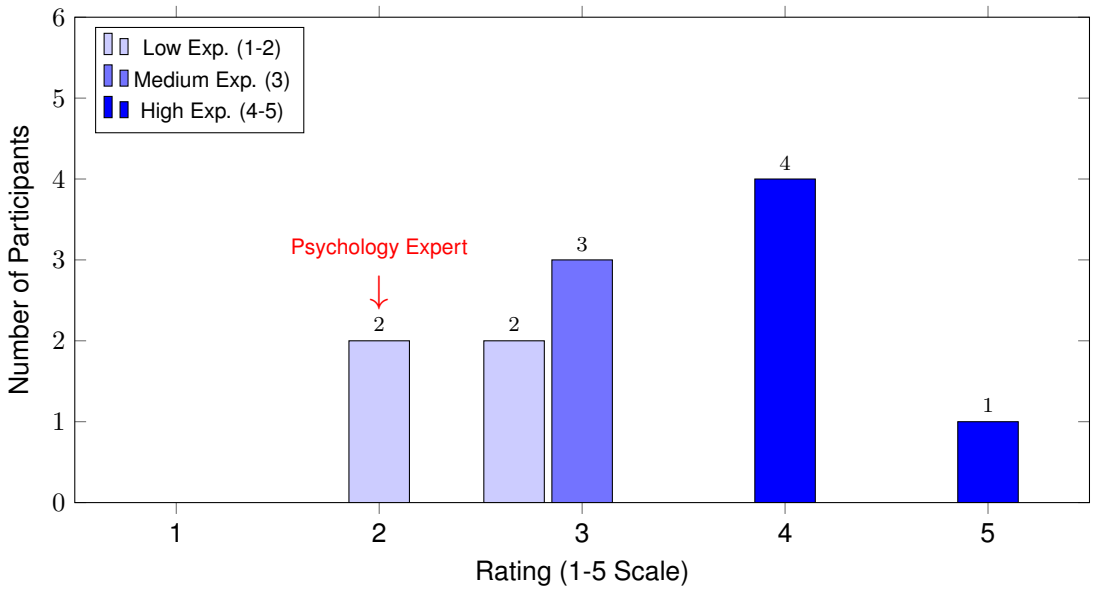


Figure 5.5: Editor ease-of-use ratings by game development experience level.

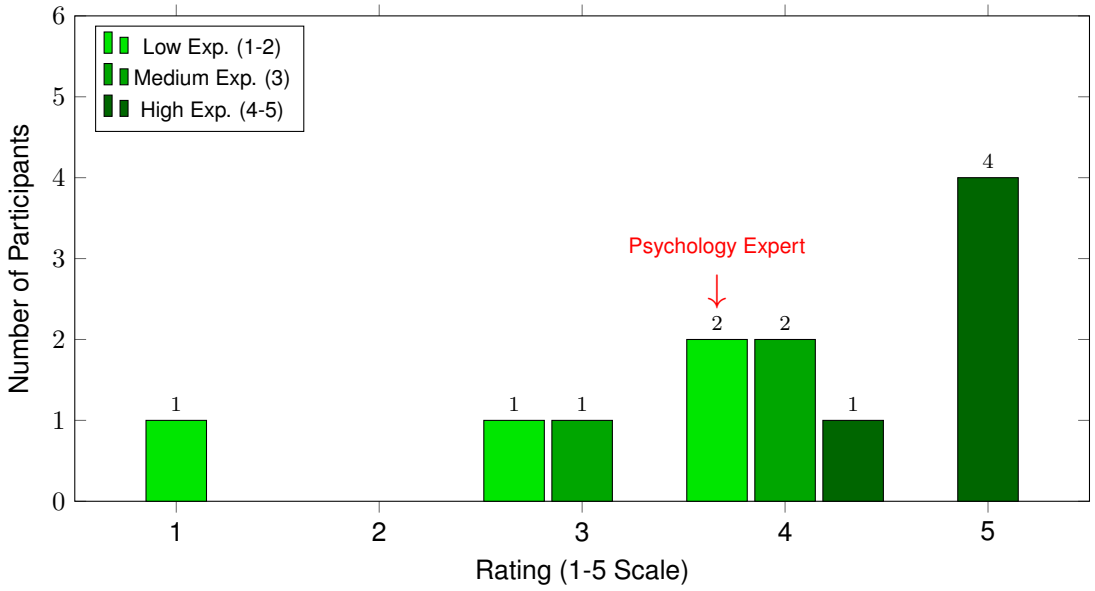


Figure 5.6: Character editing comfort ratings by game development experience level.

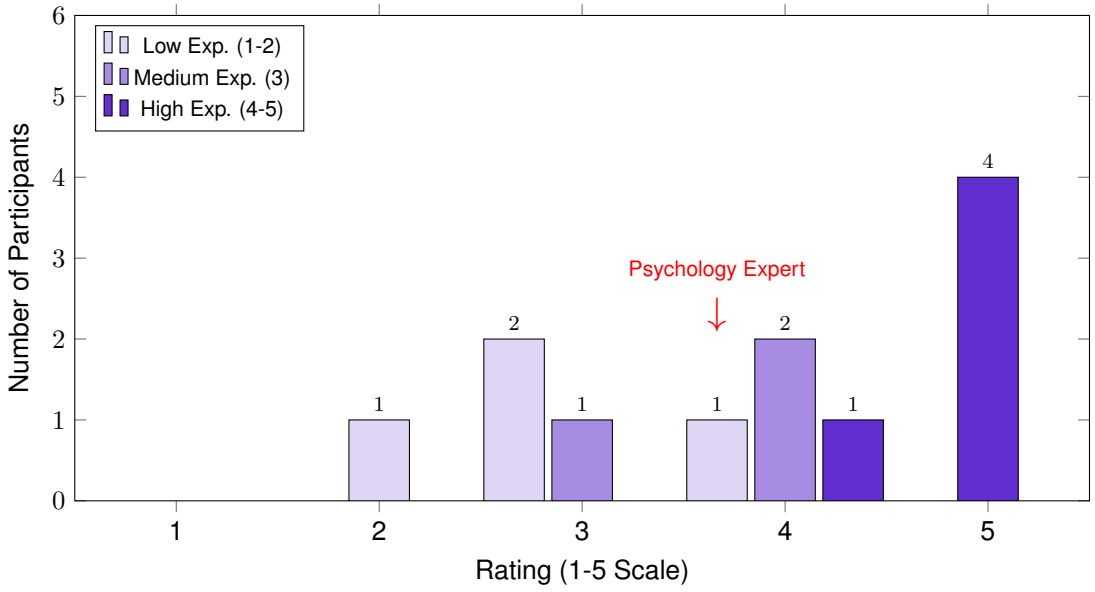


Figure 5.7: Dialogue creation comfort ratings by game development experience level.

It is important to note that the editor interface used in this evaluation was developed by a previous student as part of a separate project. Since the primary focus of this thesis was on the LLM-based dialogue generation system rather than interface design, the existing editor interface was not redesigned completely; only small adjustments were made to basic affordances and signifiers. Further concrete interface suggestions raised by participants are in Section 5.3.6 Qualitative Feedback.

Interpreting the distributions in figures 5.5, 5.7, and 5.6, a clear trend emerges: participants with

higher game development experience (ratings 4–5) tended to report better ease of use and greater comfort than those with lower experience (ratings 1–2). This likely reflects prior familiarity with editing paradigms and mental models that reduce cognitive load, while less experienced users were more affected by UI friction.

5.3.3 AI-Generated Content Performance

The evaluation of AI-generated dialogue quality revealed high adoption rates and generally positive reception of the system's content generation capabilities.

5.3.3.A AI Suggestion Adoption and Usage

- **Universal adoption**: 100% of participants (12/12) used AI-generated suggestions, with 91.67% (11/12) rating their usage as extensive (rating 5)
- **Suggestion efficiency**: 66.67% of participants (8/12) required only one AI suggestion per dialogue line, indicating high initial suggestion quality
- **Minimal editing**: Remarkably, 100% of participants (12/12) used AI suggestions without any modifications, suggesting strong alignment between generated content and user expectations

Figure 5.8 illustrates the distribution of AI suggestion usage among participants, demonstrating the system's efficiency in providing satisfactory content with minimal iteration. To contextualize the single outlier who reached 8 iterations per line, the participant clarified that this behavior persisted throughout the entire session as he actively experimented with different parameter combinations to observe the resulting outcomes, in their own words: "I was having fun and curious about the AI suggestions :)".

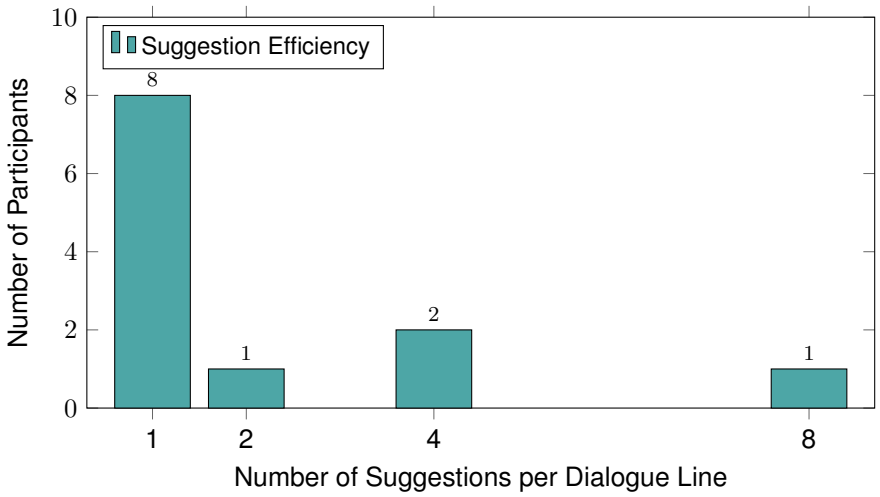


Figure 5.8: AI suggestion efficiency.

5.3.3.B Content Quality Assessment Results

Figure 5.9 presents the evaluation results for AI-generated dialogue quality across multiple dimensions:

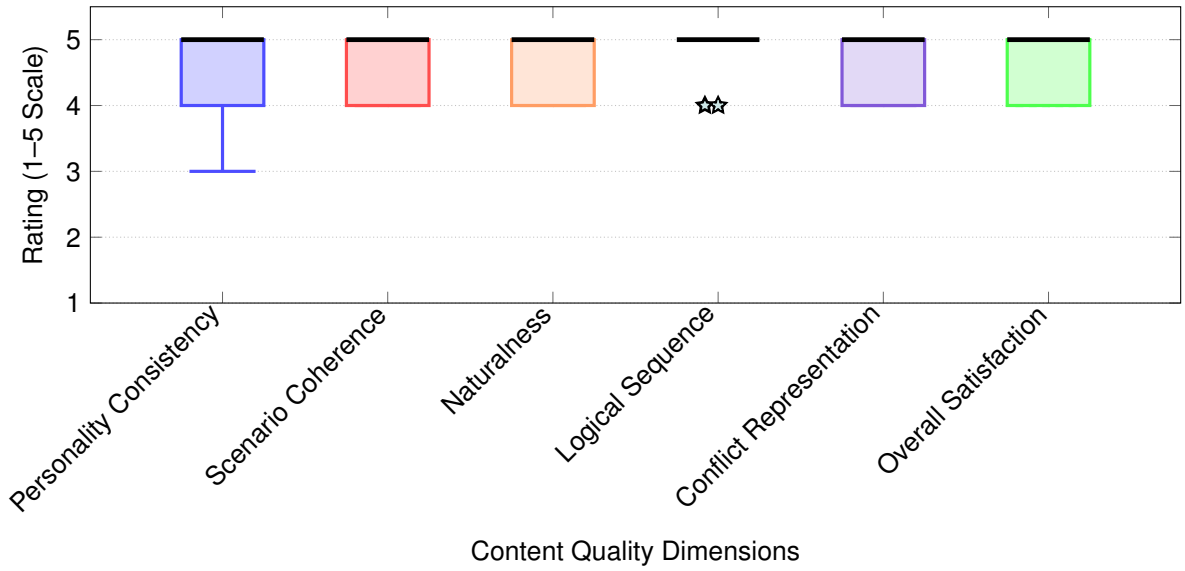


Figure 5.9: Boxplots of participant ratings for AI-generated content quality dimensions (1–5 Likert scale).

Reading the boxplots. The black line inside each box is the median; boxes span the interquartile range (IQR) (Q1–Q3) and whiskers extend up to 1.5 × IQR. In these data, all medians sit at the top of the scale (5). Only **Logical Sequence** has a fully collapsed box at 5 (Q1=median=Q3), with two lower points at 4. Only **Personality Consistency** shows a lower whisker reaching the minimum, signaling occasional neutrality.

The results demonstrate consistently high quality ratings:

- **Personality consistency**: 83.33% of participants (10/12) agreed that dialogues matched character personalities (ratings 4-5)
- **Scenario coherence**: 100% of participants (12/12) found dialogues appropriate to the context (ratings 4-5)
- **Naturalness and realism**: 100% of participants (12/12) rated dialogues as natural and believable (ratings 4-5)
- **Logical sequence**: 100% of participants (12/12) found dialogue progression coherent (ratings 4-5), with 83.33% (10/12) providing the highest rating
- **Conflict representation**: 100% of participants (12/12) agreed that dialogues effectively conveyed character relationships and conflicts (ratings 4-5)

- **Overall satisfaction**: 100% of participants (12/12) expressed satisfaction with their final dialogues (ratings 4-5)

Notably, the Psychology expert participant provided positive feedback on personality consistency, indicating validation to the psychological accuracy of the character representations from a expert's perspective. This suggests that the FFM–based personality may feel more intuitive to readers with psychological training than to participants unfamiliar with those constructs, even though trait levels were also presented as plain-language adjectives to aid understanding.

5.3.4 System Parameters

The evaluation also examined how effectively participants utilized and perceived the system's configurable parameters, which allow users to control AI suggestion generation and customize their editing experience.

5.3.4.A Parameter Adequacy Results

The system's parameter configuration capabilities were evaluated to assess whether the available control mechanisms adequately support dialogue creation workflows. Participants assessed the sufficiency of the current parameter set, which includes emotion specification, topic guidance, bullying intensity levels, and character targeting options for steering AI-generated content toward desired narrative outcomes.

Figure 5.10 presents the distribution of participant ratings regarding parameter sufficiency on the established 5-point Likert scale.

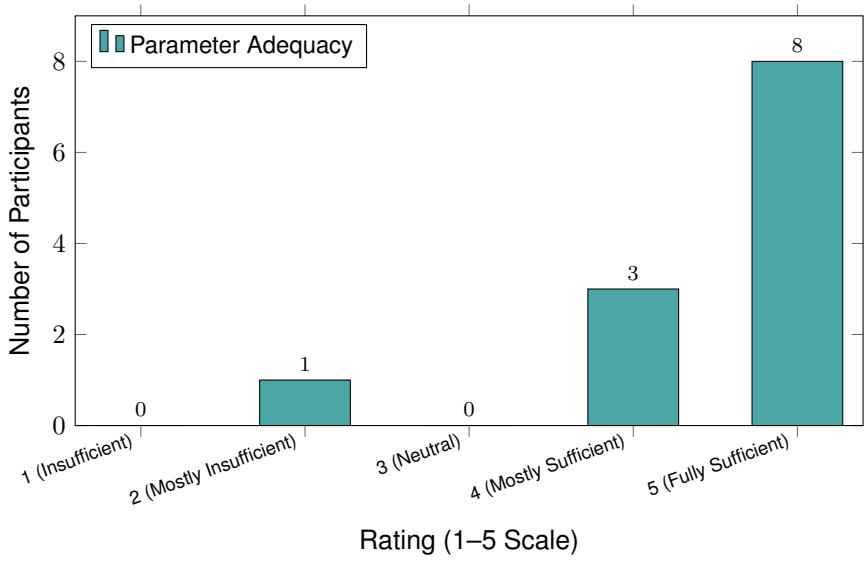


Figure 5.10: Distribution of participant ratings for parameter sufficiency in dialogue creation guidance.

The evaluation revealed strong consensus regarding parameter adequacy, with 91.67% of participants (11/12) rating the available parameters as sufficient or fully sufficient (ratings 4–5). The distribution demonstrates a clear positive skew, with the modal response at the maximum rating (5), indicating that the majority of participants found the current parameter set comprehensive for their dialogue creation needs. Only one participant expressed reservations (rating 2), while no participants selected the neutral position or expressed complete dissatisfaction.

The median rating of 5 with minimal variance suggests consistent agreement that the implemented parameter framework provides adequate granular control over AI-generated content. This finding validates the current parameter design choices and indicates that the system successfully balances control granularity with interface complexity.

5.3.5 Simulation and Final Game Experience

After completing the dialogue creation process, participants tested their created content through the integrated game simulation feature. This final evaluation phase assessed how well the system translated the authored dialogues into a playable experience and whether the simulation met user expectations.

5.3.5.A Simulation Performance Results

The simulation evaluation examined three key aspects of the final game experience: expectation alignment, dialogue coherence in gameplay context, and fidelity to the original creative intent. Figure 5.11 presents the distribution of participant ratings across these dimensions.

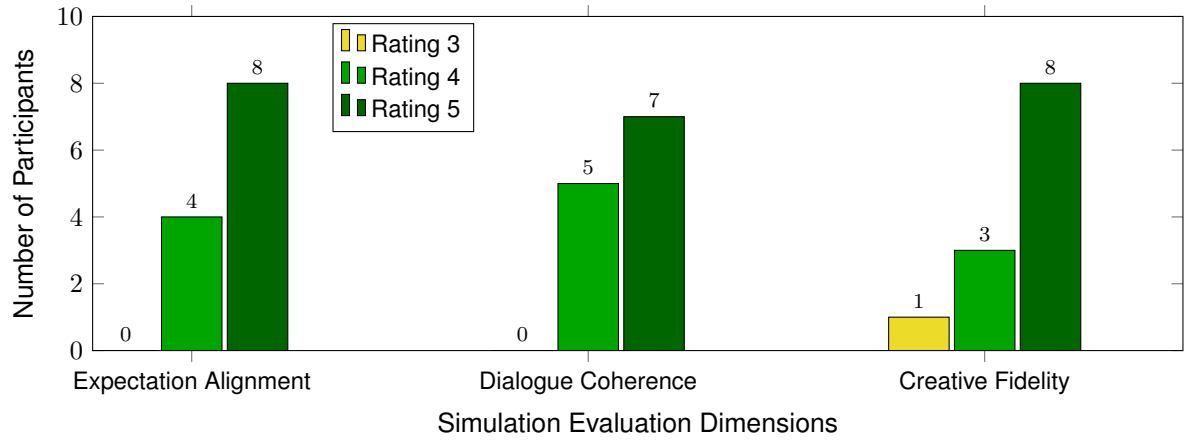


Figure 5.11: Simulation evaluation results showing participant ratings across key experience dimensions.

The simulation evaluation revealed consistently positive reception across all measured dimensions:

- **Expectation alignment**: 100% of participants (12/12) found that the simulation corresponded to their expectations (ratings 4-5), with 66.67% (8/12) providing the highest rating

- **Dialogue coherence in gameplay**: 100% of participants (12/12) rated the dialogue sequence as coherent and natural within the game context (ratings 4-5), with 58.33% (7/12) expressing complete satisfaction
- **Creative fidelity**: 91.67% of participants (11/12) agreed that the final result reflected their intended dialogues (ratings 4-5), with 66.67% (8/12) providing the highest rating and one neutral response

5.3.6 Qualitative Feedback

All participants provided comprehensive qualitative feedback through open-ended responses, offering valuable insights into both system strengths and areas requiring improvement.

System Strengths and Positive Feedback:

Participants highlighted several key advantages of the LLM-based dialogue generation system:

- **Time Efficiency and Productivity**: Multiple participants emphasized the significant time savings, with one noting "The ease of creating dialogues that would probably take 1 hour to make in just a few minutes, with just a few clicks".
- **AI Integration and Control**: Users appreciated the balance between AI assistance and human control, as expressed by a participant: "Being able to use AI and have influence over the result".
- **Creative Freedom and Flexibility**: Participants valued the system's ability to support diverse scenarios, with feedback such as "I especially liked the possibility of creating quite diverse scenarios without having to think of a complete dialogue from start to finish".
- **Context Understanding**: The AI's ability to interpret topics and maintain coherence was highly appreciated, with one participant specifically mentioning "the interpretation of topics".
- **Quality of AI Suggestions**: Several participants were positively surprised by the AI's performance, noting that "the dialogues are creative and the AI suggestions are a big positive surprise".

Areas for Improvement and Critical Feedback:

Despite the overall positive reception, participants identified several areas where the system could be improved:

- **Interface Usability Issues**: Multiple participants noted problems with interface interaction, particularly "clicking on things like 'Dialogue Name' instead of simply the text box is not intuitive at all".
- **Personality Option**: Feedback indicated that "creating the character's personality could be manual", suggesting the need for simplified character definition workflows and the optative use of the FFM.

- **Workflow Automation Requests**: Users desired more automated content generation, with suggestions for features where the user “write the first line and the AI immediately creates a full conversation”.
- **System Integration Concerns**: One participant noted the desire for broader applicability: “it could be a library and not just an engine component”.

Specific Enhancement Requests:

The qualitative feedback also revealed several concrete improvement suggestions:

- **AI Suggestion Diversity Control**: “Being able to change how similar suggestions are to each other”, this one refers specifically to the temperature parameter sent to the LLM.
- **Targeted Dialogue Generation**: “add speaking to whom” - participants requested the ability to specify which character a dialogue line is directed towards, enabling more precise conversation flow control.
- **Batch Suggestion Generation**: “immediately request multiple suggestions at once” - users wanted the capability to generate several different dialogue options simultaneously, improving workflow efficiency by providing various alternatives to choose from.
- **Suggestion History Management**: “keep a history of unused suggestions for the user to see” - participants requested a system to maintain and access previously generated suggestions, allowing them to compare and revisit alternative content options.
- **Enhanced Parameter Options**: Requests for “more emotions” and custom character-specific parameters like “being able to add that Tatiana has depression, that Patrícia gets sick easily...”.

This qualitative feedback demonstrates high user engagement and provides clear directions for future system development, balancing appreciation for current capabilities with constructive suggestions for enhancement.

5.4 Interpretation

This section analyzes the evaluation findings, discusses their implications for LLM-based dialogue generation systems, and contextualizes the results within the broader domain of AI-assisted content creation for game development.

5.4.1 Key Findings and Implications

The evaluation results reveal several important insights about the effectiveness and acceptance of LLM-based dialogue generation systems in game development contexts.

5.4.1.A System Effectiveness and User Adoption

The minimal editing requirements observed during the evaluation (100% used AI suggestions without modifications) demonstrate the system's effectiveness in generating contextually appropriate content. This finding suggests that current LLM capabilities, when properly contextualized through character personalities and scenario parameters, can successfully support creative workflows without imposing excessive post-processing overhead on users.

The efficiency metrics further support this conclusion, with 66.67% of participants requiring only a single AI suggestion per dialogue line. The average number of AI suggestions per dialogue line across all participants was 2.17, indicating efficient content generation. This efficiency level demonstrates that the system successfully captures user intent and context on the first attempt in most cases, reducing the iterative refinement typically associated with AI-generated content.

5.4.1.B Content Quality

The consistently high ratings across all content quality dimensions demonstrate that LLMs can generate dialogue content that meets professional creative standards. While logical sequence achieved the highest satisfaction rate, this dimension is generally easier to satisfy compared to more nuanced aspects of dialogue generation. More importantly, the results show that scenario coherence, naturalness, conflict representation and overall satisfaction also received perfect satisfaction rates, and personality consistency was rated positively by 83.33% of participants, reminding the rest 16.67% evaluated this field as neutral. These findings indicate that the system not only maintains narrative flow but also excels at generating contextually appropriate, believable, and character-aligned dialogue, key challenges in professional creative writing.

5.4.1.C Interface Design and Workflow Integration

The moderate usability ratings highlight the importance of interface design in AI-assisted creative tools. While the core dialogue generation functionality received strong positive feedback (75% comfortable with dialogue creation), interface usability issues prevent the system from reaching its full potential.

The qualitative feedback emphasizes specific interaction design challenges, particularly around input field affordances and visual feedback. These findings suggest that successful AI-assisted creative tools require careful attention to user experience design, not just AI model performance.

5.4.2 Theoretical and Practical Implications

5.4.2.A Implications for AI-Assisted Creativity

The results support the theoretical framework of AI as a creative collaborator rather than a replacement for human creativity. The high satisfaction rates combined with requests for enhanced control features (temperature adjustment, suggestion history) indicate that users value AI systems that amplify their creative capabilities while maintaining authorial agency. However, it is crucial that designers and creators maintain active cognitive engagement with their work, as over-reliance on AI automation risks diminishing critical thinking and creative problem-solving skills that are essential for meaningful artistic expression.

The success of personality-driven dialogue generation validates the approach of embedding structured character representations in AI content generation. The FFM integration demonstrates how psychological theories can inform AI system design to improve output quality and user satisfaction.

5.4.2.B Implications for Game Development Workflows

The time efficiency gains highlighted in participant feedback have important implications for game development pipelines. Multiple participants emphasized substantial time savings, with one noting the ability to create content "that would probably take 1 hour to make in just a few minutes." This suggests potential for substantial productivity improvements in narrative-heavy games, where dialogue writing represents a significant development bottleneck.

The system's effectiveness across different dialogue types (friendship dynamics, conflict representation, social exclusion) demonstrates versatility sufficient for complex narrative scenarios common in modern games.

5.4.2.C Implications for LLM Application Design

The evaluation validates several design principles for LLM-based creative tools:

- **Context-rich prompting**: The success of personality and scenario-driven generation demonstrates the importance of providing rich contextual information to LLMs
- **Parameter transparency**: User requests for temperature control and suggestion diversity indicate the value of exposing relevant LLM parameters to creative users
- **Workflow integration**: The positive reception of the integrated character-to-dialogue pipeline suggests that all-inclusive tool design is preferable to isolated AI features

5.4.3 Comparison with Related Work

This evaluation builds upon and extends several areas of previous research while addressing gaps identified in the related work. The findings provide new insights that advance our understanding of LLM integration in creative game development workflows.

5.4.3.A Advances Beyond Previous Psychological NPC Design

While previous work on psychological insights in NPC design (such as implementations in *The Sims* and *Persona 5*) relied on extensive manual scripting to maintain personality consistency, this thesis demonstrates that LLMs can effectively automate this process. The 83.33% satisfaction rate for personality consistency shows that embedding the FFM into LLM prompts successfully generates character-aligned dialogue without the manual workload traditionally required. This addresses the challenge identified in related work regarding the difficulty of "maintaining coherence and variety across scenarios" in traditional approaches.

These results are consistent with academic literature on believable characters and personality-driven agents, which operationalizes trait models and social context for virtual behavior and dialogue [9, 13]. They also align with recent evidence that LLMs can simulate FFM traits and produce personality-conditioned behaviors [17, 18]. Finally, our evaluation dimensions echo established practices for assessing narrative quality and coherence in academic settings [25].

5.4.3.B LLM Integration Insights

Previous work on LLMs for narrative generation has highlighted recurring challenges: maintaining narrative consistency, controlling tone and personality, and ensuring alignment with design constraints. This thesis addresses these issues through structured character conditioning (personalities, relationships, and scenario parameters) and context-rich prompting. Within the scope of our study, these techniques appear to mitigate consistency and control issues, at least for short dialogue scenarios, though their effectiveness in larger or long-form narrative contexts remains an open question.

The preference for human-AI collaboration over full automation aligns with industry practices noted in related work. This thesis validates this approach empirically, showing that users value maintaining "influence over the result" while benefiting from AI efficiency.

5.4.3.C Novel Contributions to Game Development Tool Design

While previous work on AI integration in game development tools focused primarily on asset creation and general workflow support, this thesis specifically addresses dialogue generation within game editors. The finding that comprehensive tool design (integrated character-to-dialogue pipeline) is preferable to

isolated AI features provides new design principles for future AI-assisted game development tools. The high content quality ratings demonstrate the effectiveness of this approach, suggesting that dialogue generation represents a particularly well-suited application for current LLM capabilities.

5.5 Limitations and Potential Biases

This section provides a brief summary of validity considerations to contextualize the reported results. A complete analysis of study limitations and the corresponding future research directions is presented in Chapter 6.

5.5.1 Validity Summary

- **Internal validity**: The researcher's presence and study framing may have introduced expectancy effects; a novelty effect is possible given first-time exposure to AI-assisted dialogue tools. Mitigation included a standardized protocol and systematic observation notes.
- **External validity**: Convenience sampling (N=12, predominantly Computer Engineering students aged 19–24) and a single visual-novel scenario in a Portuguese high-school context limit generalization across populations, genres, and cultures.
- **Construct validity**: Likert instruments were adapted for this context, and constructs like "personality consistency" and "creative control" may not fully capture their broader theoretical scope.

5.5.2 Mitigating Factors

- Mixed-methods triangulation (quantitative + qualitative) to balance subjective measures.
- Standardized evaluation protocol for consistency across participants.
- Multiple evaluation dimensions (usability, content quality, parameters, overall experience) for a holistic view.
- Transparent reporting to enable readers to assess applicability to their contexts.

For the full enumeration of limitations, methodological constraints, and implications for future work, see Chapter 6.

6

Conclusion and Future Work

Contents

6.1 Introduction

This thesis has explored the integration of LLMs into game development workflows to assist designers in creating personality-driven, contextually appropriate NPC dialogues during the development phase. Throughout this research journey, we have moved from identifying the fundamental challenges in dynamic dialogue generation to developing, implementing, and evaluating a comprehensive solution that demonstrates the potential of AI-assisted creative tools in game development.

The motivation for this work emerged from the recognition that traditional dialogue creation methods, whether through scripted trees or rule-based systems, impose significant limitations on both creative expression and development efficiency. While these approaches have served the industry for decades, they struggle to accommodate the growing demand for rich, varied, and believable character interactions that modern players expect. The advent of powerful LLMs presented an opportunity to address these limitations for crafting meaningful narratives.

Our investigation began with establishing the context of this work in Chapter 1, which presented the motivation, problem framing, and research questions guiding this work. We then surveyed the landscape of existing research and theoretical foundations in Chapter 2, analyzing previous approaches to dialogue systems, psychological insights in NPC design, and the emerging applications of LLMs in narrative generation. This background work revealed both the potential of current technologies and the gaps that our research aimed to address.

The core contribution of this thesis, presented in Chapter 3, is a novel system architecture that integrates LLM-based dialogue generation directly into a game editor environment. Unlike previous approaches that focus on runtime generation, our solution emphasizes a human-in-the-loop workflow during the design phase, enabling designers to leverage AI assistance while maintaining full editorial control over the final content. The system incorporates structured personality modeling through the FFM framework, contextual relationship management, and configurable dialogue parameters to produce character-aligned, contextually appropriate dialogue suggestions.

All of the evaluation methodology and experiment is detailed in Chapter 4, and the empirical validation of our approach, detailed in Chapter 5, involved a comprehensive mixed-methods evaluation with 12 participants from the academic community. The evaluation demonstrated strong evidence for the effectiveness of our approach, with participants completing all tasks, high satisfaction with generated content, and time savings in dialogue creation workflows.

This final chapter synthesizes the key findings of the research, articulating its main contributions to both academic understanding and practical application, while outlining potential directions for future investigation. It begins with a summary of the overall research trajectory and a restatement of the specific problems addressed, followed by an analysis of the contributions across technical, empirical, and practical dimensions. The chapter then provides concise answers to the research questions that

guided this work, reflects on the insights gained throughout the development and evaluation process, and concludes with a roadmap for future research and system enhancement within this evolving domain.

The implications of this work extend beyond the specific technical contributions to broader questions about the role of AI in creative processes. Our findings contribute to the growing body of evidence that AI can serve as an effective creative collaborator when properly integrated into human workflows, while also highlighting the critical importance of maintaining human agency and control in creative endeavors. As the capabilities of LLMs continue to advance and their integration into creative tools becomes more widespread, the principles and findings presented in this thesis provide valuable guidance for researchers, developers, and practitioners working at the intersection of AI and creative expression.

6.2 Thesis Summary

This section provides a concise overview of the research conducted in this thesis, summarizing the problem context, proposed solution, and implementation approach that together form the foundation of our contributions to AI-assisted dialogue generation in game development.

6.2.1 Problem Context and Motivation

As established in Chapter 1 (Motivation and Context) and revisited in Chapter 2 (Related Work), prescripted dialogue trees and rule-based systems limit consistent personality expression and development efficiency in dialogue-heavy games. In brief, these constraints motivate an editor-time, human-in-the-loop approach where LLMs assist designers while preserving editorial control. This chapter summarizes that framing, and positions our solution (Chapter 3) as a response to those validated challenges.

6.2.2 Proposed Solution Overview

This thesis proposed and implemented a novel approach to LLM integration that addresses the limitations of traditional methods.

The solution is guided by a few high-level principles:

- **Editor-time focus**: eliminate runtime latency and enable thorough editorial review
- **Human-in-the-loop**: designers accept, modify, or reject suggestions
- **Context-driven prompting**: condition generation on traits, relationships, emotions, and scenario
- **Safety and traceability**: keep prompts/outputs auditable for iteration

6.2.3 Implementation Approach

We implemented a modular system integrated in the editor with focus on two main components: a **Character Editor** that encodes FFM-based personalities and relationships, and a **Dialogue Editor** that exposes concise control parameters and invokes the LLM to propose lines aligned with the current context. Prompt construction, integration layer behavior, persistence, and UI specifics are described in Chapter 3. This implementation demonstrates that AI assistance can be embedded in creative workflows without compromising designer agency.

6.3 Key Contributions

This thesis makes significant contributions across multiple dimensions of research and practice in AI-assisted creative tools for game development. The contributions span technical innovation, empirical validation, and practical application, each addressing specific gaps in the current state of knowledge and practice.

6.3.1 Technical Contributions

6.3.1.A Novel LLM Integration Architecture

The primary technical contribution of this work is the development of a novel architecture for integrating LLMs into game development workflows.

Key technical innovations include:

- **Structured Prompt Engineering Framework**: Development of a comprehensive prompt construction system that dynamically incorporates character personality profiles, relationship contexts, emotional states, and scenario parameters to generate contextually appropriate dialogue suggestions
- **Personality-LLM Bridge**: Creation of an automated system that translates FFM personality trait combinations into natural language descriptions suitable for LLM conditioning, enabling consistent character-aligned dialogue generation
- **Multi-parameter Dialogue Control**: Implementation of a sophisticated parameter system allowing designers to control dialogue generation through speaker selection, emotional state, social dynamics (including bullying levels), and thematic topics
- **Asynchronous Processing Architecture**: Design of a background processing system that maintains interface responsiveness during LLM API calls while providing user feedback and timeout handling

6.3.1.B Character Relationship Modeling

This thesis contributes a novel approach to modeling and maintaining character relationships within AI-assisted dialogue generation systems. The relationship management system handles bidirectional relationship updates and maintains social network consistency across complex character casts.

The relationship modeling framework includes:

- **Bidirectional Relationship Consistency**: Automatic propagation of relationship changes across character pairs to maintain logical consistency
- **Context-Aware Dialogue Conditioning**: Integration of relationship information into LLM prompts to ensure generated dialogue reflects established social dynamics

6.3.2 Empirical Contributions

6.3.2.A Comprehensive Evaluation Methodology

This research contributes a robust mixed-methods evaluation framework specifically designed for assessing AI-assisted creative tools in game development contexts. The methodology combines quantitative usability metrics with qualitative user experience assessment to provide comprehensive validation of system effectiveness.

The evaluation framework includes:

- **Usability Metrics**: Development of evaluation criteria specifically adapted for AI-assisted creative workflows, including suggestion adoption rates and editing frequency
- **Content Quality Assessment**: Establishment of multi-dimensional quality evaluation encompassing personality consistency, contextual coherence, naturalness, logical sequence, and conflict representation
- **Mixed-Methods Integration**: Systematic combination of quantitative measurements with qualitative feedback analysis to capture both measurable performance and subjective user experience

6.3.2.B Empirical Validation of Human-AI Collaboration

The evaluation conducted in this thesis provides concrete empirical evidence for the effectiveness of human-AI collaboration in creative contexts.

Key empirical findings include:

- **High Adoption Rates**: 100% of participants successfully used AI suggestions

- **Efficiency Validation**: Average of 2.17 AI suggestions per dialogue line, with majority of participants requiring only one suggestion per line
- **Quality Confirmation**: High satisfaction rates across all content quality dimensions, demonstrating that AI-generated content meets professional creative standards
- **Creative Control Preservation**: 100% satisfaction with final dialogues combined with user positive feedback

6.3.3 Practical Contributions

6.3.3.A Framework for AI-Assisted Game Development

This thesis provides a practical framework for implementing AI assistance in game development workflows that can be adapted across different development contexts and tools. The framework emphasizes maintainable human-AI collaboration while preserving the creative agency essential for meaningful artistic expression.

The practical framework includes:

- **Design Principles for Creative AI Tools**: Establishment of guidelines for balancing automation with human control, parameter transparency, and workflow integration
- **Implementation Guidelines**: Concrete recommendations for integrating LLM capabilities into existing game development pipelines without disrupting established processes
- **User Experience Patterns**: Identification of effective interaction patterns for AI-assisted creative tools, including suggestion presentation, editing workflows, and feedback mechanisms

6.4 Research Questions Answered

This section provides comprehensive answers to the two research questions that guided this thesis, as established in Chapter 1. Each answer is supported by specific evidence from our evaluation study and contextualizes the findings within the broader implications for AI-assisted creative tools.

6.4.1 RQ1: LLM Effectiveness in Character-Aligned Dialogue Generation

Research Question 1: *Can LLMs effectively generate contextually appropriate and character-aligned dialogue content?*

Answer: Yes, our evaluation provides empirical evidence that LLMs can effectively generate high-quality, character-aligned dialogue content when properly contextualized through structured personality models and scenario parameters.

6.4.1.A Supporting Evidence

The evaluation study demonstrates LLM effectiveness across multiple quality dimensions:

- **Scenario Coherence**: 100% of participants rated AI-generated dialogues as appropriate to the given context (ratings 4-5), indicating complete success in contextual understanding and application
- **Personality Consistency**: 83.33% of participants agreed that dialogues matched predefined character personalities (ratings 4-5), with the remaining 16.67% providing neutral assessments, demonstrating strong character alignment
- **Naturalness and Realism**: 100% of participants found the dialogues natural and believable (ratings 4-5), validating the LLM's ability to produce human-like conversational content
- **Logical Sequence**: 100% of participants rated dialogue progression as coherent (ratings 4-5), with 83% providing the highest rating, indicating superior conversational flow
- **Conflict Representation**: 100% of participants agreed that dialogues effectively conveyed character relationships and social dynamics (ratings 4-5)

6.4.1.B Professional Psychological Feedback

This study included feedback from a single participant with psychological training. While informative, this input does not constitute formal validation. Her comments suggested that several generated lines were broadly consistent with intended personality traits and revealed no obvious contradictions with basic psychological principles. However, given the single-expert perspective and the short-form dialogues used in our evaluation, these observations should be interpreted as preliminary and context-limited.

6.4.1.C Technical Factors Contributing to Success

Several technical design decisions proved crucial for achieving this effectiveness:

- **Multi-Parameter Prompting**: The combination of character personalities, relationships, emotional states, and scenario topics created comprehensive context for generation
- **Editor-time Positioning**: Removing runtime constraints allowed for more sophisticated prompt engineering and content quality optimization

6.4.1.D Cross-Domain Validation

The system demonstrated effectiveness across diverse dialogue types and social scenarios:

- **Friendship Dynamics**: Successful generation of supportive, collaborative dialogue reflecting positive relationships
- **Conflict Scenarios**: Appropriate handling of tension, disagreement, and social exclusion dynamics
- **Complex Social Situations**: Effective representation of multi-character interactions involving sensitive topics like bullying and peer pressure

6.4.2 RQ2: Implementation Success Factors

Research Question 2: *What are the key factors for successful implementation of LLM-based dialogue generation systems in game development workflows?*

Answer: Our evaluation identified several critical success factors that determine the effectiveness of LLM integration into game development workflows, spanning technical architecture, user experience design, and workflow integration considerations.

6.4.2.A Technical Architecture Factors

- **Context-Rich Prompting**: The success of personality and scenario-driven generation demonstrates the critical importance of providing comprehensive contextual information to LLMs rather than relying on generic prompts
- **Modular System Design**: The separation of character definition and dialogue generation components enabled independent optimization and easier maintenance
- **Asynchronous Processing**: Background processing with timeout handling maintained interface responsiveness during LLM API calls, preventing workflow disruption
- **Structured Data Integration**: Automatic conversion of personality traits and relationships into natural language descriptions proved essential for consistent AI conditioning

6.4.2.B User Experience Design Factors

- **Parameter Transparency**: User requests for temperature control and suggestion diversity indicate the value of exposing relevant LLM parameters to creative users rather than hiding system complexity

- **Workflow Integration**: The positive reception of the integrated character-to-dialogue pipeline suggests that comprehensive tool design is preferable to isolated AI features
- **Feedback and Control**: Users valued having "influence over the result", indicating that successful AI creative tools must maintain human agency throughout the process

6.4.2.C Process Integration Factors

- **Editor-time Focus**: Positioning AI assistance during development rather than runtime proved crucial for maintaining quality control and creative oversight
- **Quality Assurance Integration**: The ability to review and refine AI suggestions before finalizing content addressed concerns about appropriateness and narrative alignment

6.4.2.D Identified Areas for Improvement

The evaluation also revealed specific areas requiring attention for optimal implementation:

- **Interface Usability**: Specific interaction design challenges around input field affordances and visual feedback require careful attention
- **Enhanced Parameter Control**: Users requested additional controls including suggestion diversity (temperature), targeted dialogue generation ("speaking to whom"), and batch suggestion capabilities
- **Suggestion Management**: Requests for suggestion history and comparison features indicate the value of supporting iterative refinement workflows
- **Workflow Automation**: User interest in features like "write the first line and AI creates full conversation" suggests opportunities for optional higher-level automation

These findings provide actionable guidance for future implementations and highlight the importance of balancing AI capabilities with human creative agency in successful AI-assisted creative tools.

6.5 Lessons Learned

This section reflects on the insights gained throughout the research process, from initial system design through implementation and evaluation. These lessons provide valuable guidance for future research in AI-assisted creative tools and highlight both the successes and challenges encountered during this work.

6.5.1 What Worked Well

6.5.1.A Editor-time Integration Strategy

The decision to focus on the development stage rather than runtime integration proved to be one successful aspect of this research:

- **Quality Control Preservation**: By generating suggestions during development, designers maintained complete editorial oversight over final content, addressing industry concerns about inappropriate or off-brand AI output
- **Creative Workflow Compatibility**: The approach integrated naturally into existing development processes without requiring fundamental changes to established creative workflows

6.5.1.B Structured Personality Integration

The integration of the FFM with LLM conditioning exceeded expectations in several ways:

- **Designer Accessibility**: Most non-experts could effectively use the personality system without deep theoretical knowledge, though the evaluation showed some participants without psychology backgrounds occasionally found abstract trait combinations challenging to interpret
- **Content Quality**: The personality-driven approach consistently produced dialogue that felt authentic and character-appropriate, as confirmed by evaluation results

6.5.2 Unexpected Findings

6.5.2.A User Exploration Behavior

The evaluation revealed interesting patterns in how users interacted with AI suggestions:

- **Curiosity-Driven Usage**: One participant requested 8 AI suggestions per line “because I was having fun and curious about the AI suggestions”, highlighting the engaging nature of the technology
- **Immediate Acceptance**: The high rate of suggestion acceptance without editing (100%) was higher than anticipated, suggesting strong initial quality
- **Quality Expectations**: Participants demonstrated high expectations for AI-generated content quality, validating the importance of sophisticated prompt engineering

6.5.2.B Workflow Integration Insights

Several aspects of workflow integration provided unexpected insights:

- **Seamless Adoption**: Users adapted to AI-assisted workflows more quickly than expected, with most expressing comfort after minimal training
- **Creative Process Enhancement**: Rather than disrupting creative flow, AI assistance appeared to enhance ideation and reduce creative blocks
- **Quality Consistency**: The consistency of AI-generated content quality across different users and scenarios exceeded initial expectations

6.5.3 Research Process Reflections

The iterative approach to system development and evaluation proved essential:

- **Iterative Self-Testing**: Regular personal testing of the system throughout development helped identify and address usability issues and improve AI suggestion quality
- **Technical Validation**: Regular testing identified API reliability issues early in the development process
- **Scope Management**: Iterative development helped maintain focus on core functionality while identifying future enhancement opportunities

6.6 Limitations and Future Work

While this research has made contributions to the field of AI-assisted dialogue generation, it is important to acknowledge the limitations of the current work and identify promising directions for future research.

6.6.1 Study Limitations

6.6.1.A Sample Size and Generalizability

The evaluation was conducted with a limited participant pool, which constrains the generalizability of findings:

- **Participant Count**: With 12 participants, the study provides valuable insights but may not capture the full diversity of potential user behaviors and preferences

- **Domain Expertise**: Participants were primarily recruited from academic and technical backgrounds, potentially limiting insights into how the system might perform with broader creative communities
- **Cultural Context**: The evaluation was conducted within a specific cultural and linguistic context, raising questions about cross-cultural applicability
- **Experience Levels**: The range of participant experience with both game development and AI tools may have influenced adoption patterns and usage behaviors

6.6.1.B Technical Scope Constraints

Several technical limitations influenced the system's capabilities and evaluation outcomes:

- **LLM Model Dependency**: The system's performance is inherently limited by the capabilities and biases of the underlying language models used
- **Time Performance**: The 5-15 second generation times, while acceptable for evaluation, may not meet all production workflow requirements
- **Interface Maturity**: Interface usability issues may have influenced user experience and satisfaction ratings generation approach.

6.6.1.C Evaluation Methodology Constraints

The chosen evaluation approach, while comprehensive, had inherent limitations:

- **Artificial Scenarios**: Evaluation tasks, though carefully designed, may not fully replicate the complexity and pressure of real-world game development contexts
- **Short-term Assessment**: The evaluation captured immediate user responses but did not assess long-term adoption patterns or workflow integration effects
- **Subjective Measurement Challenges**: Quantifying creative quality and user satisfaction relies heavily on subjective assessments that may vary significantly between individuals
- **Control Group Absence**: The study design did not include control groups working without AI assistance, limiting comparative analysis possibilities

6.6.2 Future Research Directions

The findings and limitations of this work suggest several promising avenues for future research that could significantly advance the field of AI-assisted creative tools.

6.6.2.A Enhanced Personality Modeling

Future work could substantially improve the sophistication and accuracy of personality representation:

- **Multi-dimensional Personality Models**: Exploring integration of additional personality frameworks beyond the FFM
- **Dynamic Personality Evolution**: Developing systems that can model how character personalities evolve throughout a narrative, reflecting character development arcs
- **Contextual Personality Expression**: Research into how personality traits should be expressed differently across various social contexts and situations within games
- **Cultural Personality Variations**: Investigating how personality models should be adapted to reflect different cultural backgrounds and social norms

6.6.2.B Advanced Dialogue Generation Techniques

Several technical improvements could enhance the quality and sophistication of generated dialogue:

- **Multi-character Conversation Management**: Extending the system to handle complex multi-character dialogues with appropriate turn-taking and interaction dynamics
- **Emotional Arc Integration**: Developing techniques to ensure dialogue reflects and advances emotional storylines across longer narrative sequences
- **Genre-specific Adaptation**: Researching how dialogue generation should be customized for different game genres (e.g., horror, comedy, drama, fantasy)
- **Voice and Style Consistency**: Improving long-term consistency in character voice, speech patterns, and linguistic quirks across extensive dialogue sequences

6.6.2.C Evaluation Methodology Advancement

Future research should develop more sophisticated evaluation approaches for AI-assisted creative tools:

- **Comparative Framework Development**: Establishing standardized benchmarks and comparison methodologies for AI-assisted creative tools across different domains
- **Real-world Integration Studies**: Evaluating AI assistance tools within actual game development projects to understand production-environment impacts
- **Cross-cultural Validation**: Replicating evaluations across different cultural contexts to ensure broad applicability of findings

- **Expert Panel Validation**: Incorporating multiple psychology experts in the evaluation process to provide more robust assessment of personality-consistent dialogue generation

6.6.2.D System Integration and Scalability

Future research should address practical deployment and scalability challenges:

- **Production Workflow Integration**: Developing seamless integration approaches for existing game development pipelines and professional creative tools
- **Collaborative AI Assistance**: Researching how AI tools can support team-based creative processes and collaborative dialogue development
- **Performance Optimization**: Investigating techniques to reduce generation times and improve system responsiveness for creative workflows
- **Cost-effective Deployment**: Developing strategies to make AI-assisted creative tools accessible to independent developers and smaller creative teams

6.6.2.E Technical Research Priorities

Several technical research directions emerge as particularly promising:

- **Adaptive Prompting**: Developing systems that automatically adjust prompt strategies based on user preferences and content quality feedback
- **Context-Aware Generation**: Improving techniques for maintaining narrative consistency across complex, branching dialogue structures
- **Multi-modal Integration**: Exploring how visual, audio, and textual context can be combined to generate more contextually appropriate dialogue
- **Intelligent Suggestion Timing**: Researching optimal moments for AI intervention in creative workflows to maximize assistance while minimizing disruption

The research presented in this thesis establishes a foundation for these future developments while demonstrating the immediate viability and value of AI-assisted dialogue generation in creative contexts. The combination of technical innovation and comprehensive evaluation provides a robust starting point for the continued evolution of AI-assisted creative tools.

6.7 Final Reflections

As this research journey concludes, it is important to reflect on the broader implications of our findings for the intersection of AI and creative expression in game development and beyond.

6.7.1 The Promise of Human-AI Creative Collaboration

The results of this thesis demonstrate that thoughtfully designed AI assistance can enhance human creativity rather than replace it. Our participants consistently reported that AI suggestions sparked new creative directions while never feeling that their agency was compromised. This finding challenges common concerns about AI diminishing human creativity and instead supports a vision of AI as a creative collaborator that amplifies human capabilities.

The key to this successful collaboration lies in the preservation of human agency throughout the creative process. By positioning AI as a suggestion provider rather than a decision maker, we maintained the essential human elements of creative judgment, artistic vision, and emotional understanding that are fundamental to meaningful storytelling. This approach offers a model for AI integration across creative domains where human creativity and AI capabilities can complement each other productively.

6.7.2 Implications for Game Development Practice

From a practical perspective, this research demonstrates that AI-assisted dialogue generation is not only technically feasible but immediately beneficial for game development workflows. The significant time savings reported by participants, combined with high satisfaction rates and maintained creative control, suggest that such tools could have substantial positive impacts on game development efficiency and creative output quality.

The success of our human-in-the-loop approach also validates the importance of editor-time AI assistance over runtime generation for creative applications. This finding has implications for how the game development industry might approach AI integration more broadly, suggesting that editor-time assistance could be more valuable than runtime automation for many creative tasks.

6.7.3 Looking Forward

As LLMs continue to advance in capability and accessibility, the opportunities for creative AI assistance will only expand. The framework established in this thesis, emphasizing human agency, contextual understanding, and seamless workflow integration, provides a foundation for future developments in this space.

The game development industry, with its combination of technical sophistication and creative demands, serves as an ideal testing ground for advanced AI-assisted creative tools. The principles and findings from this research can inform the development of similar tools across other creative domains, from interactive fiction to educational content creation.

6.7.4 Closing Thoughts

This thesis began with the observation that traditional dialogue creation methods in game development, while functional, impose significant limitations on creative expression and development efficiency. Through careful research, implementation, and evaluation, we have demonstrated that AI-assisted dialogue generation offers a viable path forward that addresses these limitations while preserving the human creativity that makes games meaningful.

The most sophisticated AI capabilities are only as valuable as their practical application in real human workflows, and success depends on understanding and supporting human creative processes rather than attempting to replace them.

As we look toward the future of AI-assisted creative tools, the work presented in this thesis offers both practical solutions for immediate application and a research foundation for continued innovation. The intersection of AI and human creativity holds tremendous potential for enhancing how we create, share, and experience interactive narratives. By maintaining focus on human agency, creative control, and meaningful collaboration, we can harness this potential to support and amplify the remarkable creative capabilities of human designers and storytellers.

The dialogue between humans and AI in creative contexts has only just begun, and this thesis contributes one voice to that ongoing conversation. The future of creative expression lies not in choosing between human creativity and AI capability, but in discovering how they can work together to create experiences that neither could achieve alone.

Bibliography

[1] S. Sharma and V. Sharma, "How ai transforms play: The evolution of artificial intelligence in video games," *Media and AI: Navigating*, p. 151, 2024. [Online]. Available: https://www.researchgate.net/profile/Aaqib-Butt/publication/381229239_Media_and_AI_Navigating_The_Future_of_Communication/links/6662a3bba54c5f0b9451c383/Media-and-AI-Navigating-The-Future-of-Communication.pdf#page=151

[2] R. Gallotta, G. Todd, M. Zammit, S. Earle, A. Liapis, J. Togelius, and G. N. Yannakakis, "Large language models and games: A survey and roadmap," *arXiv preprint arXiv:2402.18659*, 2024.

[3] H. Panwar, "The npc ai of the last of us: a case study," *arXiv preprint arXiv:2207.00682*, 2022.

[4] T.-Y. Ennabili, "A comparison of traditional game design vs. ai-driven game design," 2023, bachelor's thesis, Häme University of Applied Sciences. [Online]. Available: https://www.theseus.fi/handle/10024/816173

[5] C. Strong and M. Mateas, "Talking with npcs: Towards dynamic generation of discourse structures," in *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, vol. 4, no. 1, 2008, pp. 114–119.

[6] L. M. Csepregi, "The effect of context-aware llm-based npc conversations on player engagement in role-playing video games," Master's Thesis, Aalborg University, 2021. [Online]. Available: https://projekter.aau.dk/projekter/files/536738243/The_Effect_of_Context_aware_LLM_based_NPC_Dialogues_on_Player_Engagement_in_Role_playing_Video_Games.pdf

[7] J. Huang, "Generating dynamic and lifelike npc dialogs in role-playing games using large language models," Bachelor's Thesis, LUT University, 2024. [Online]. Available: https://lutpub.lut.fi/handle/10024/167809

[8] J. Fraser, I. Papaioannou, and O. Lemon, "Spoken conversational ai in video games: Emotional dialogue management increases user engagement," in *Proceedings of the 18th international conference on intelligent virtual agents*, 2018, pp. 179–184. [Online]. Available: https://dl.acm.org/doi/10.1145/3267851.3267896

[9] K. Isbister, *Better game characters by design: A psychological approach*. CRC Press, 2022.

[10] R. R. McCrae and O. P. John, "An introduction to the five-factor model and its applications," *Journal of personality*, vol. 60, no. 2, pp. 175–215, 1992.

[11] C. J. Soto and J. J. Jackson, "Five-factor model of personality," *Journal of Research in Personality*, vol. 42, pp. 1285–1302, 2013.

[12] L. J. Klinkert, S. Buongiorno, and C. Clark, "Driving generative agents with their personality," *arXiv preprint arXiv:2402.14879*, 2024. [Online]. Available: https://arxiv.org/abs/2402.14879

[13] J. You, "Comprehensive believable non-player characters creation and management tools for emergent gameplay," Ph.D. dissertation, University of Western Ontario, 2009. [Online]. Available: https://ir.lib.uwo.ca/digitizedtheses/3837

[14] C. Bailey and M. Katchabaw, "An emergent framework for realistic psychosocial behaviour in non-player characters," in *Proceedings of the 2008 Conference on Future Play: Research, Play, Share*. ACM, 2008, pp. 17–24.

[15] A. C. Tavares, R. R. da Mota, and W. Melo, "Self-reflection in games-the representation of the individuation process in celeste and persona 2: Innocent sin," in *Simpósio Brasileiro de Jogos e Entretenimento Digital (SBGames)*. SBC, 2021, pp. 198–207.

[16] J. Georgeson and C. Child, "Npcs as people, too: The extreme ai personality engine," *arXiv preprint arXiv:1609.04879*, 2016. [Online]. Available: https://arxiv.org/abs/1609.04879

[17] B. Han, D. Kwon, S. Lin, K. Shrestha, and J. Gratch, "Can llms generate behaviors for embodied virtual agents based on personality traits?" *arXiv preprint arXiv:2508.21087*, 2025.

[18] A. Sorokovikova, N. Fedorova, S. Rezagholi, and I. P. Yamshchikov, "Llms simulate big five personality traits: Further evidence," *arXiv preprint arXiv:2402.01765*, 2024.

[19] A. E. Poropat, "A meta-analysis of the five-factor model of personality and academic performance," *Psychological Bulletin*, vol. 135, no. 2, pp. 322–338, 2009. [Online]. Available: https://doi.org/10.1037/a0014996

[20] D. B. Samuel and T. A. Widiger, "A meta-analytic review of the relationships between the five-factor model and dsm-iv-tr personality disorders: A facet level analysis," *Clinical Psychology Review*, vol. 28, no. 8, pp. 1326–1342, 2008. [Online]. Available: https://doi.org/10.1016/j.cpr.2008.07.002

[21] J. Costa, Paul T. and R. R. McCrae, *Revised NEO Personality Inventory (NEO-PI-R) and NEO Five-Factor Inventory (NEO-FFI) Professional Manual*. Odessa, FL: Psychological Assessment Resources, 1992.

[22] A. Furnham, “The big five versus the big four: The relationship between the myers–briggs type indicator (mbti) and neo-pi five-factor model of personality,” *Personality and Individual Differences*, vol. 21, no. 2, pp. 303–307, 1996. [Online]. Available: https://doi.org/10.1016/0191-8869(96)00033-5

[23] M. C. Ashton and K. Lee, “Empirical, theoretical, and practical advantages of the hexaco model of personality structure,” *Personality and Social Psychology Review*, vol. 11, no. 2, pp. 150–166, 2007. [Online]. Available: https://doi.org/10.1177/1088868306294907

[24] G. Petri and C. G. von Wangenheim, “How to evaluate educational games: a systematic literature review,” *Journal of Universal Computer Science*, vol. 22, no. 7, pp. 992–1021, 2016.

[25] M. O. Riedl and R. M. Young, “Narrative planning: Balancing plot and character,” *Journal of Artificial Intelligence Research*, vol. 39, pp. 217–268, 2010. [Online]. Available: https://jair.org/index.php/jair/article/view/10669

[26] J. T. Tan and R. W. Picard, “Affective computing and intelligent interaction,” in *Proceedings of the Second International Conference on Affective Computing and Intelligent Interaction (ACII 2007)*. Berlin, Germany: Springer, 2007, pp. 1–2.

[27] H. Martins, H. Marques, and C. Martinho, “A serious game platform to train teachers on cyber-bullying prevention and response,” in *2024 IEEE conference on games (CoG)*. IEEE, 2024, pp. 1–4.

[28] E. Brown and P. Cairns, “A grounded investigation of game immersion,” in *CHI’04 Extended Abstracts on Human Factors in Computing Systems*. ACM, 2004, pp. 1297–1300. [Online]. Available: https://dl.acm.org/doi/10.1145/985921.986048

[29] C. Jennett, A. L. Cox, P. Cairns, S. Dhoparee, A. Epps, T. Tijs, and A. Walton, “Measuring and defining the experience of immersion in games,” *International Journal of Human-Computer Studies*, vol. 66, no. 9, pp. 641–661, 2008. [Online]. Available: https://doi.org/10.1016/j.ijhcs.2008.04.004

[30] H. L. O’Brien and E. G. Toms, “What is user engagement? a conceptual framework for defining user engagement with technology,” *Journal of the American Society for Information Science and Technology*, vol. 59, no. 6, pp. 938–955, 2008. [Online]. Available: https://doi.org/10.1002/asi.20801

[31] K. Poels, Y. A. de Kort, and W. A. IJsselsteijn, “D3.3: Game experience questionnaire: Development of a self-report measure to assess the psychological impact of digital games,” 2007, unpublished manuscript. [Online]. Available: https://research.tue.nl/en/publications/d33-game-experience-questionnaire-development-of-a-self-report-me

[32] J. H. Brockmyer, C. M. Fox, K. A. Curtiss, E. McBroom, K. M. Burkhart, and J. N. Pidruzny, "The development of the game engagement questionnaire: A measure of engagement in video game-playing," *Journal of experimental social psychology*, vol. 45, no. 4, pp. 624–634, 2009.

[33] W. Ribbens, "Perceived game realism: A test of three alternative models," *Cyberpsychology, Behavior, and Social Networking*, vol. 16, no. 1, pp. 31–36, 2013. [Online]. Available: https://doi.org/10.1089/cyber.2012.0212

[34] R. B. Rubin, P. Palmgreen, and H. E. Sypher, "Perceived realism scale," in *Communication Research Measures*. Routledge, 2020, pp. 282–285.

[35] C. M. Rose, "Realistic dialogue engine for video games," Master's Thesis, The University of Western Ontario, 2014. [Online]. Available: https://ir.lib.uwo.ca/etd/2652/

[36] T. Ashby, B. K. Webb, G. Knapp, J. Searle, and N. Fulda, "Personalized quest and dialogue generation in role-playing games: A knowledge graph-and language model-based approach," in *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. ACM, 2023, pp. 1–20. [Online]. Available: https://dl.acm.org/doi/10.1145/3544548.3581441

[37] S. Buongiorno, L. J. Klinkert, T. Chawla, Z. Zhuang, and C. Clark, "Pangea: Procedural artificial narrative using generative ai for turn-based video games," *arXiv preprint arXiv:2404.19721*, 2024. [Online]. Available: https://arxiv.org/abs/2404.19721

[38] N. Nananukul and W. Wongkamjan, "What if red can talk? dynamic dialogue generation using large language models," 2024, arXiv preprint arXiv:2407.20382.

[39] DAIR.AI, "Prompt engineering guide," https://www.promptingguide.ai/, 2024, accessed: 2025-10-07.

[40] W. Liang, C. Zhao, C. Deng *et al.*, "Deepseek-v3 technical report: Architecture, training, and performance of a 671b parameter mixture-of-experts language model," *arXiv preprint*, vol. 2412.19437, 2024, v2. [Online]. Available: https://arxiv.org/abs/2412.19437

[41] D. Xiao, C. Gao, Z. Luo, C. Liu, and S. Shen, "Can llms assist computer education? an empirical case study of deepseek," *arXiv preprint*, vol. 2504.00421, 2025. [Online]. Available: https://arxiv.org/abs/2504.00421

[42] R. K. Anam, "Prompt engineering and the effectiveness of large language models in enhancing human productivity," *arXiv preprint arXiv:2507.18638*, 2025.

[43] R. Liu, A. Rashid, I. Kobyzev, M. Rezagholizadeh, and P. Poupart, "Attribute controlled dialogue prompting," *arXiv preprint arXiv:2307.05228*, 2023.

[44] F. Calimeri, S. Germano, G. Ianni, F. Pacenza, S. Perri, and J. Zangari, “Integrating rule-based ai tools into mainstream game development,” in *Proceedings of the International Joint Conference on Rules and Reasoning (RuleML+RR)*. Cham: Springer, 2018, pp. 310–317.

[45] D. Angilica, G. Greco, and G. Ianni, “Tight integration of artificial intelligence in game development tools,” Ph.D. Thesis, Università della Calabria, Cosenza, Italy, 2020.

[46] J. P. Zagal, N. Tomuro, and A. Shepitsen, “Natural language processing in game studies research: An overview,” *Simulation & Gaming*, vol. 43, no. 3, pp. 356–373, 2012. [Online]. Available: https://journals.sagepub.com/doi/full/10.1177/1046878111422560

[47] M. O. Riedl and A. Zook, “Ai for game production,” in *2013 IEEE Conference on Computational Intelligence and Games (CIG)*. IEEE, 2013, pp. 1–8. [Online]. Available: https://ieeexplore.ieee.org/document/6633663

[48] P. Vijayakumar, M. Pyingkodi, S. Devi *et al.*, “Comparative analysis of ai chatbot for assessing gemini ai, deepseek ai, and qwen ai via openrouter api integration,” in *2025 6th International Conference on Data Intelligence and Cognitive Informatics (ICDICI)*. IEEE, 2025, pp. 01–08.

[49] R. Ciesla, *Game Development with Ren’Py*. Springer, 2019.

[50] P. T. Costa and R. R. McCrae, *Revised NEO Personality Inventory (NEO PI-R) and NEO Five-Factor Inventory (NEO-FFI) professional manual*. Odessa, FL: Psychological Assessment Resources, 1992.

[51] ——, “Domains and facets: Hierarchical personality assessment using the revised neo personality inventory,” *Journal of Personality Assessment*, vol. 64, no. 1, pp. 21–50, 1995.

A

Implementation Code

This appendix contains the key implementation algorithms for the LLM-based dialogue generation system presented in Chapter 4, organized according to the system architecture workflow: Editor Interface, Prompting & Control Module, and LLM Integration Layer.

A.1 Editor Interface Implementation

The Editor Interface consists of two main components: the Character Editor for personality and relationship management, and the Dialogue Editor for parameter configuration.

A.1.1 Character Editor Algorithm

Listing A.1: Character Editor - Personality and Relationship Management

```
1 FUNCTION get_personality_summary(personality_dict):
2         Converts Big Five traits (0-2 scale) into natural language description

3     IF personality_dict is empty:
```

```
        RETURN  Default.

    # Extract trait values: extroverted, friendly, responsible, anxious,
        creative
    traits = EXTRACT_TRAIT_VALUES(personality_dict)

    # Generate personality facets from trait combinations
    facets = []
    FOR each_pair IN TRAIT_COMBINATIONS(traits):
        IF both_traits_are_non_neutral:
            facet = SELECT_RANDOM_FACET(trait_combination_mapping[pair])
            facets.APPEND(facet)

    # Limit to 5 facets and return description
    RETURN JOIN(LIMIT(facets, 5),  ,  )

FUNCTION apply_mutual_relationships(char_id, relationships):
       Manages bidirectional character relationships
    # Remove old relationships from other characters
    FOR other_char IN all_characters:
        DELETE other_char.relationships[char_id]

    # Apply new mutual relationships
    FOR target_id, relationship_type IN relationships:
        target_character.relationships[char_id] = relationship_type

FUNCTION create_character():
       Main character creation workflow
    character = NEW_CHARACTER(name, visual_properties)
    character.personality = DEEP_COPY(char_personality)
    character.relationships = DEEP_COPY(char_relationships)

    apply_mutual_relationships(character.id, char_relationships)
    STORE_CHARACTER(character)
```

A.1.2 Dialogue Editor Algorithm

Listing A.2: Dialogue Editor - Parameter Configuration and LLM Integration

```
FUNCTION trigger_dialogue_suggestion(speaker_id, emotion_id, referenced_chars
    ):
       Initiates background LLM dialogue generation
    SHOW_LOADING_SCREEN()

    # Prepare context data
    context = {
         participants : dialogue_chars,
         history : format_dialogue_history(),
         scenario : dialogue_scenario,
         bullying_level : dialogue_bullying_level
    }

    # Start background thread for LLM request
    START_BACKGROUND_THREAD(
        FUNCTION: generate_dialogue_with_context(speaker_id, emotion_id,
            context),
        CALLBACK: update_dialogue_field,
        TIMEOUT: 30_seconds
    )

FUNCTION configure_dialogue_parameters():
       Handles dialogue parameter selection
    parameters = {
         speaker : SELECT_FROM_CHARACTERS(),
         emotion : SELECT_FROM_EMOTIONS(),  # 7 emotions available
         referenced_chars : SELECT_MULTIPLE_CHARACTERS(),
         bullying_level : SELECT_FROM_LEVELS(),  # None/Low/Medium/High
         scenario : INPUT_TEXT_FIELD()
    }
    RETURN parameters

FUNCTION add_dialogue_line(speaker_id, emotion_id, speech_text):
       Validates and adds dialogue line to conversation
    IF all_parameters_valid(speaker_id, emotion_id, speech_text):
        line = [speaker_id, emotion_id, speech_text]
        local_lines.APPEND(line)
```

```
36          INCREMENT line_counter
```

A.2 Prompting & Control Module Implementation

This module constructs comprehensive prompts by combining character data, scene context, and dialogue parameters into structured LLM requests.

A.2.1 Prompt Construction Algorithm

Listing A.3: Prompt Construction - Context Integration

```
1  FUNCTION prepare_dialogue_prompt(speaker_id, emotion_id, context_data):
2          Builds structured prompt with character and scene context
3      prompt =  Generate dialogue in European Portuguese for character:\n\n
4
5      # Section 1: Speaker Information
6      speaker_info = GET_SPEAKER_INFO(speaker_id)
7      prompt +=  Speaker:   + speaker_info.name +  \n
8      prompt +=  Emotion:   + GET_EMOTION_NAME(emotion_id) +  \n
9      prompt +=  Personality:   + GET_PERSONALITY_SUMMARY(speaker_info.
           personality) +  \n
10
11     # Section 2: Context and Relationships
12     prompt +=  \nContext:\n
13     prompt +=  Setting: High school environment\n
14     IF context_data.scenario:
15         prompt +=  Topic:   + context_data.scenario +  \n
16
17     # Add relationship context
18     relationships = GET_SPEAKER_RELATIONSHIPS(speaker_id)
19     IF relationships:
20         prompt +=  Relationships:   + FORMAT_RELATIONSHIPS(relationships) +
               \n
21
22     # Section 3: Conversation Parameters
23     IF context_data.participants:
```

```
24          prompt += "Present: " + JOIN_NAMES(context_data.participants) + "\n"
25      IF context_data.referenced_chars:
26          prompt += "Mentions: " + JOIN_NAMES(context_data.referenced_chars) +
                "\n"
27      IF context_data.bullying_level != None :
28          prompt += "Tone: " + DESCRIBE_BULLYING_LEVEL(context_data.
                bullying_level) + "\n"
29
30      # Section 4: Dialogue History
31      IF context_data.history:
32          prompt += "\nPrevious dialogue:\n" + context_data.history + "\n"
33
34      # Section 5: Output Requirements
35      prompt += "\nGenerate ONE line of natural Portuguese teenage dialogue."
36
37      RETURN prompt
38
39  FUNCTION format_dialogue_history(lines):
40          Formats recent dialogue for context
41      history =
42      FOR line IN LAST_N_LINES(lines, 3):
43          speaker_name = GET_SPEAKER_NAME(line.speaker_id)
44          emotion = GET_EMOTION_NAME(line.emotion_id)
45          history += speaker_name + "(" + emotion + "): \" + line.text + "\\n"

46      RETURN history
```

A.3 LLM Integration Layer Implementation

This component handles communication with external LLM services, including error handling, retry logic, and local fallback mechanisms.

A.3.1 Core Integration Algorithm

Listing A.4: LLM Integration - API Communication and Fallback

```
FUNCTION generate_dialogue(prompt):
        Main LLM integration with OpenRouter API and local fallback
    # Configuration
    API_URL =  https://openrouter.ai/api/v1/chat/completions
    MODEL =  deepseek/deepseek-chat-v3-0324:free
    TIMEOUT = 15
    LOCAL_FALLBACK_URL =  http://localhost:11434/api/generate

    TRY:
        # Build API request
        request_data = {
             model : MODEL,
             messages : [
                { role :  system ,  content :  Generate dialogue for visual
                    novels },
                { role :  user ,  content : prompt}
            ],
             max_tokens : 150,
             temperature : 1
        }

        # Execute primary API call
        response = HTTP_POST(API_URL, request_data, TIMEOUT)
        result = PARSE_RESPONSE(response)

        IF result.success:
            RETURN EXTRACT_DIALOGUE(result)
        ELSE:
            RETURN CALL_LOCAL_FALLBACK(prompt)

    EXCEPT (TimeoutError, NetworkError, APIError):
        RETURN CALL_LOCAL_FALLBACK(prompt)

FUNCTION call_local_fallback(prompt):
        Local Ollama fallback for offline operation
    TRY:
        fallback_data = {
             model :  llama3.2:1b ,
```

```
                prompt : prompt,
                stream : false
            }

            response = HTTP_POST(LOCAL_FALLBACK_URL, fallback_data, 10)
            result = PARSE_LOCAL_RESPONSE(response)

            IF result.success:
                RETURN EXTRACT_DIALOGUE(result)
            ELSE:
                RETURN  Fallback failed: Unable to generate dialogue

        EXCEPT:
            RETURN  Error: Both primary and fallback services unavailable

FUNCTION parse_response(response):
        Extracts dialogue content from API response
        IF response.contains( choices ) AND response.choices.length > 0:
            dialogue = response.choices[0].message.content.STRIP()
            IF dialogue.length > 0:
                RETURN SUCCESS(dialogue)

        RETURN ERROR( No valid dialogue in response )
```

A.3.2 Implementation Overview

This concise implementation captures the essential workflow of the LLM integration:

- **API Configuration**: Direct connection to OpenRouter using DeepSeek's free model
- **Request Structure**: System prompt for context plus user dialogue prompt
- **Response Processing**: Extraction of generated text from API response with basic validation
- **Error Handling**: Graceful failure modes for common network and API issues
- **Local Fallback**: Ollama integration with smaller local model (llama3.2:1b) for resilience and offline operation

A.4 Evaluation Materials

This section contains the supporting materials used during the experimental evaluation described in Chapter 4.

A.4.1 Step-by-Step Test Guide

The comprehensive step-by-step test guide provided to participants during the evaluation sessions is presented below. This guide ensured consistent testing procedures across all participants and provided detailed instructions for system navigation and task completion.

Listing A.5: Test Guide

```
GUIÃO DE TESTE DO EDITOR DE JOGO
================================================
1. INTRODUÇÃO
================================================
Bem-vindo ao teste da minha tese! Este guião vai guiá-lo/a na exploração
do editor de jogo e na criação de diálogos com base num cenário definido.
Está a utilizar um editor de Novel Games (um tipo de jogo interativo focado
em histórias contadas através de texto, imagens e diálogos), criado para
ajudar designers a construir diálogos de forma intuitiva com o apoio da
inteligência artificial, no entanto, este editor ainda é um
protótipo inicial imperfeito. A sua tarefa vai consistir em experimentar
o editor, ao editar 3 personagens do jogo e criar 3 situações de diálogo
(de acordo com o cenário que defini) para depois experimentar o
minijogo resultante com os diálogos que criou (note que o editor
está todo em inglês, mas o jogo será em português)! Para isso vou
ajudá-lo/a passo a passo...
================================================
2. CONTEXTO DO CENÁRIO
================================================
A base do jogo já foi criada por mim, como a história, todas as
personagens e áreas de jogo. O cenário do jogo passa-se numa
escola secundária, onde se está a preparar a viagem de finalistas
ao Algarve, organizada pela Estrela (delegada de turma,
responsável e empática). A Cármen não pode ir para esse destino
pois os pais dizem ser muito longe e acusa Estrela de favorecer a
sua amiga Isabel que tanto quer ir para o Algarve. Tatiana
```

(tímida e passiva) é alvo de piadas por Patrícia, Manuela e
Cármen já faz muito tempo. Para se vingar, Tatiana no passado
criou um perfil falso para enganar a Patrícia e depois mandou lhe
mensagens ofensivas, Nando, atual namorado da Patrícia não gostou
de ouvir esta situação, e quer defendê-la. Rui e Estrela tentam apoiar
Tatiana pois não gostam de injustiças nem do bullying que anda a ocorrer.

==========

3. INTERFACE DO EDITOR

==========

BOTÕES PRINCIPAIS:

- NEW - Criar novos objetos (Personagens, Diálogos, Áreas)
- LIBRARY - Ver e editar objetos existentes

IMPORTANTE: Todas as personagens e áreas já estão criadas.
Existem 3 personagens incompletas para editar.

==========

4. EDITAR PERSONAGENS (Estrela, Patrícia, Rui)

==========

PASSOS:

1. Ir a Library -> Selecionar personagem
2. Focar no lado direito: Personality e Relationships
3. Usar modelo Big Five (5 traços principais):
   - Extroversão
   - Amabilidade
   - Conscienciosidade
   - Neuroticismo
   - Abertura
4. Atribuir Low , Neutral ou High a cada fator
5. Clicar Refresh se não gostar da descrição gerada
6. Adicionar ligações sociais (amizade, namoro, conflito, etc.)

==========

5. CRIAR DIÁLOGOS

==========

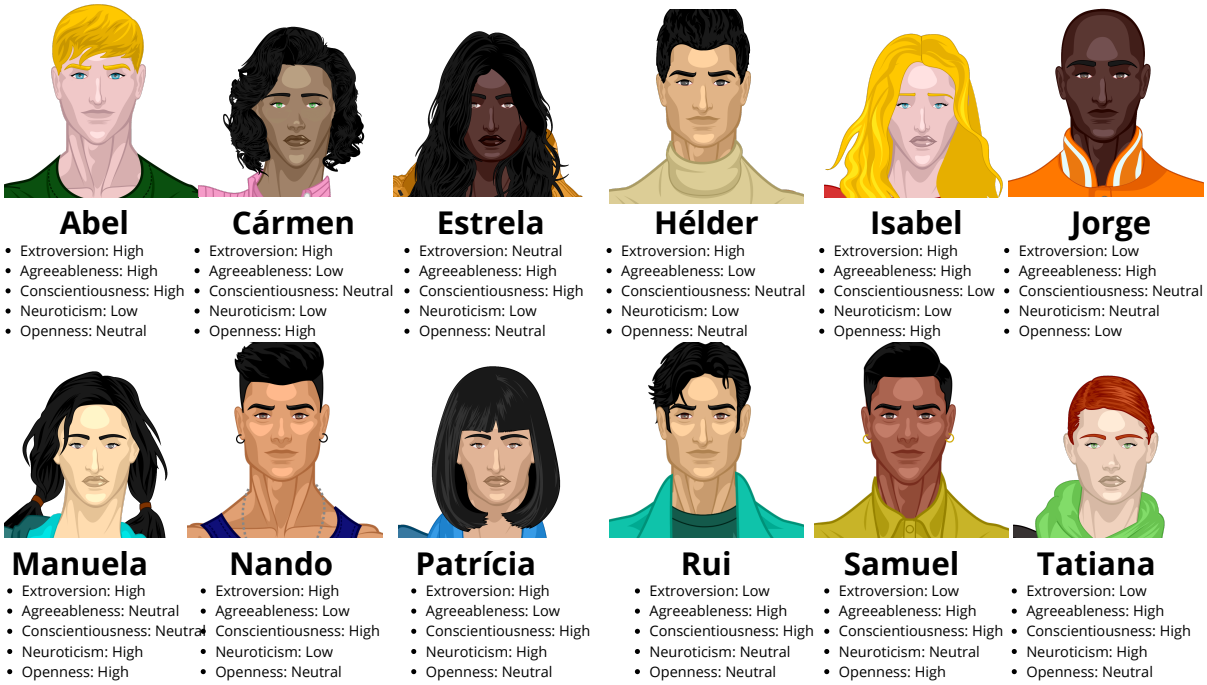
WORKFLOW:

1. NEW -> Dialogue
2. Dar nome ao diálogo (ex: Diálogo1)
3. Selecionar Participants
4. Dialogue Lines -> Add Line

```
5. Definir speaker + escolher emoção:
   (Neutro, Raiva, Tristeza, Medo, Felicidade, Desilusão, Surpresa)
6. Configurar campos opcionais:
   - Speaking of
   - Bullying Level
   - Scenario/Topic
7. Escrever fala no retângulo azul
8. Pedir sugestões à IA e editar conforme necessário
9. Confirmar -> Repetir para mais falas -> CREATE

DIÁLOGOS REQUERIDOS:

DIÁLOGO 1:
• Participantes: Tatiana, Isabel, Estrela
• Tópico: Viagem de finalistas
• Objetivo: Mostrar amizade Isabel-Estrela e insegurança de Tatiana

DIÁLOGO 2:
• Participantes: Abel, Jorge, Samuel, Rui
• Tópico: Livre
• Objetivo: Testar interações naturais baseadas nas ligações de Rui

DIÁLOGO 3:
• Participantes: Cármen, Manuela, Patrícia, Nando
• Tópico: Conversa sobre Tatiana
• Objetivo: Mostrar exclusão social e cumplicidade entre agressores
==============================================
6. TESTAR O JOGO
==============================================
1. Clicar em  Simulate
2. Clicar novamente para começar
3. Usar setas vermelhas para navegar pela escola
4. Clicar nas sombras para ver diálogos criados

Muito obrigado pela disponibilidade!
```

A.4.2 Character Information and Scenario Context

The character relationship diagram and personality reference sheet provided as supporting materials for the predefined scenario are included below.



Abel
- Extroversion: High
- Agreeableness: High
- Conscientiousness: High
- Neuroticism: Low
- Openness: Neutral

Cármen
- Extroversion: High
- Agreeableness: Low
- Conscientiousness: Neutral
- Neuroticism: Low
- Openness: High

Estrela
- Extroversion: Neutral
- Agreeableness: High
- Conscientiousness: High
- Neuroticism: Low
- Openness: Neutral

Hélder
- Extroversion: High
- Agreeableness: Low
- Conscientiousness: Neutral
- Neuroticism: Low
- Openness: Neutral

Isabel
- Extroversion: High
- Agreeableness: High
- Conscientiousness: Low
- Neuroticism: Low
- Openness: High

Jorge
- Extroversion: Low
- Agreeableness: High
- Conscientiousness: Neutral
- Neuroticism: Neutral
- Openness: Low

Manuela
- Extroversion: High
- Agreeableness: Neutral
- Conscientiousness: Neutral
- Neuroticism: High
- Openness: High

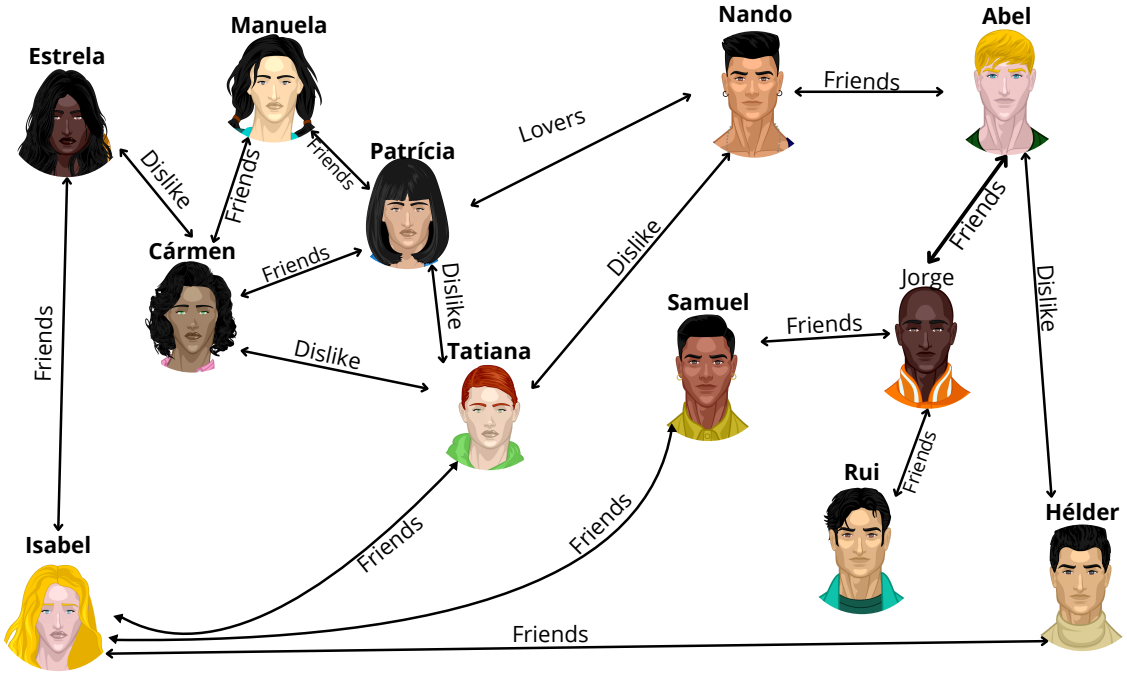
Nando
- Extroversion: High
- Agreeableness: Low
- Conscientiousness: High
- Neuroticism: Low
- Openness: Neutral

Patrícia
- Extroversion: High
- Agreeableness: Low
- Conscientiousness: High
- Neuroticism: High
- Openness: Neutral

Rui
- Extroversion: Low
- Agreeableness: High
- Conscientiousness: High
- Neuroticism: Neutral
- Openness: Neutral

Samuel
- Extroversion: Low
- Agreeableness: High
- Conscientiousness: High
- Neuroticism: Neutral
- Openness: High

Tatiana
- Extroversion: Low
- Agreeableness: High
- Conscientiousness: High
- Neuroticism: High
- Openness: Neutral



A.4.3 Test Questionnaire

The test questionnaire administered after the evaluation session to assess system performance across multiple dimensions.

Listing A.6: Post-Test Questionnaire - System Evaluation Assessment

```
POS-TESTE - QUESTIONARIO DE AVALIACAO DO SISTEMA

==============================================
SECCAO 1: USABILIDADE E INTERFACE
==============================================

Indique o seu grau de concordancia com as seguintes afirmacoes,
usando a escala:

1 - Discordo totalmente
2 - Discordo parcialmente
3 - Nem concordo nem discordo
4 - Concordo parcialmente
5 - Concordo totalmente

Foi facil utilizar o editor.
[ ] 1  [ ] 2  [ ] 3  [ ] 4  [ ] 5

Senti-me confortavel a navegar e criar dialogos.
[ ] 1  [ ] 2  [ ] 3  [ ] 4  [ ] 5

Senti-me confortavel a navegar e editar personagens.
[ ] 1  [ ] 2  [ ] 3  [ ] 4  [ ] 5

A interface pareceu-me flexivel e adaptavel as minhas decisoes.
[ ] 1  [ ] 2  [ ] 3  [ ] 4  [ ] 5

Se quiser dar algum feedback em relacao a usabilidade da interface:
-----------------------------------------------

==============================================
SECCAO 2: INTELIGENCIA ARTIFICIAL E GERACAO DE DIALOGOS
```

```
==============================================

Usei as sugestoes de dialogos geradas pela IA.
[ ] 1  [ ] 2  [ ] 3  [ ] 4  [ ] 5

Se usou sugestoes da IA e decidiu editar algumas,
quais foram as razoes para essa edicao?
[ ] A fala nao fazia sentido no contexto do dialogo
[ ] A emocao nao correspondia a situacao
[ ] A linguagem nao estava natural
[ ] A personagem nao falaria daquela forma
[ ] A sugestao era demasiado generica
[ ] Fiquei satisfeito/nao precisei de editar
[ ] Outro: --------------------------------

Os dialogos gerados pela IA estavam de acordo com as personalidades
das personagens.
[ ] 1  [ ] 2  [ ] 3  [ ] 4  [ ] 5

Os dialogos estavam coerentes com o cenario.
[ ] 1  [ ] 2  [ ] 4  [ ] 5

Os dialogos atingiram um bom nivel de realismo e naturalidade.
[ ] 1  [ ] 2  [ ] 3  [ ] 4  [ ] 5

As falas sugeridas seguiram uma sequencia logica e coerente
com as anteriores.
[ ] 1  [ ] 2  [ ] 3  [ ] 4  [ ] 5

Os dialogos ajudaram a transmitir o contexto e os conflitos
entre as personagens.
[ ] 1  [ ] 2  [ ] 3  [ ] 4  [ ] 5

Fiquei satisfeito/a com os dialogos finais criados.
[ ] 1  [ ] 2  [ ] 3  [ ] 4  [ ] 5

Se discordou com a afirmacao anterior, explique porque:
-------------------------------------------------
```

```
================================================
SECCAO 3: PARAMETROS E PERSONALIZACAO
================================================

Os parametros disponiveis (emocao, topico, nivel de bullying, etc.)
foram suficientes para orientar a criacao dos dialogos.
[ ] 1  [ ] 2  [ ] 3  [ ] 4  [ ] 5

Que outros parametros seriam uteis de acrescentar?
 ------------------------------------------------

================================================
SECCAO 4: SIMULACAO E RESULTADO FINAL
================================================

A simulacao final correspondeu as minhas expectativas.
[ ] 1  [ ] 2  [ ] 3  [ ] 4  [ ] 5

A sequencia das falas no jogo era coerente e natural.
[ ] 1  [ ] 2  [ ] 3  [ ] 4  [ ] 5

O resultado final refletiu os dialogos que quis criar.
[ ] 1  [ ] 2  [ ] 3  [ ] 4  [ ] 5

================================================
SECCAO 5: OPINIAO GERAL
================================================

O que mais gostou na ferramenta?
 ------------------------------------------------

O que melhoraria ou mudaria?
 ------------------------------------------------

Comentarios adicionais:
 ------------------------------------------------
```