Faculty of Science, Engineering, and Computing

School of Computer Science and Mathematics

Kingston University London

CI7800 Digital Media Final Project - Interim Report

MSc GAME DEVELOPMENT (PROGRAMMING)

Leveraging Large Language Models for Dynamic Game Narratives

Prajwal Shetty Vijaykumar - K2371155

Body of Creative Work
Email: k2371155@kingston.ac.uk

Supervisor: Prof. Vasileios Argyriou
Email: vasileios.argyriou@kingston.ac.uk

Release date: July 2024

Abstract:

This project proposes an alternative approach to the development of games of certain genres like role-playing games (RPGs) by integrating the Large Language Models (LLMs) into their pipelines. The goal here is to create a detective-style adventure game where the overall narration becomes responsive to the player's actions dynamically in real-time. Their choices will significantly impact the story's progression and outcome. With a combination of powerful modern-day game engines like Unreal Engine 5 and advanced cutting-edge LLMs such as GPT-4o or Claude 3.5, the game features an AI-driven dialogue system, adaptive storylines, and unique NPC behaviors with their personalities. The research explores the potential of LLMs to enhance player immersion, replayability, and engagement, and reshape interactive storytelling. The report outlines the technical implementation, including API integration, resource management, and current challenges such as cost optimization and maintaining narrative consistency. This project seeks to demonstrate the future possibilities of personalized, dynamic gaming experiences by pushing the boundaries of AI-enhanced game design.

1. Introduction:

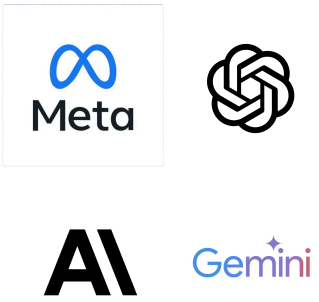
The modern-day story-driven, role-playing games have evolved into complex, sprawling franchises. While these carefully crafted experiences often result in highly satisfying outcomes for players, as seen in titles like Uncharted (PlayStation, n.d.), and Red Dead Redemption (Games, n.d.), they also come with some drawbacks. Firstly, the development of such games is an incredibly time-consuming and labor-intensive process. As game scopes continue to expand, development teams face tremendous pressure to create ever-larger and more immersive experiences, pushing the boundaries of what's feasible within the current production scale. Secondly, despite the substantial scale of these games, the overall outcome always stays consistent across the millions of players who buy these games. While this consistency allows developers to maintain control over the gameplay experience, it can also limit the potential for more diverse and personalized adventures.

With the rise of large language models (LLMs), there is an opportunity to revolutionize the whole development process. These systems have reached a level of sophistication where they can understand and respond to many of the natural human languages, including English, with near real-time response speed. The generated replies are also contextually aware and smart, opening up new possibilities for player-game world interactions. By leveraging LLMs, we can begin to break free from the constraints of predefined paths in game design. Players can now experience dynamic game flow and NPC interactions, potentially leading to storylines and endings that even the game developers themselves couldn't have anticipated. There have been works on using LLMs to explore both narration (Rao et al., 2024) and dialogues (Kalbiyev, 2022) in games before, the report will also explore the possible improvements over the previous approaches.

2. State of the art:

There are mainly two major aspects the project depends on, the first being the current state of large language models (LLMs) and the second, their integration into the current game engines and other related development pipelines.

2.1 State of Large Language Models:



The rapid progress in artificial intelligence (AI), driven by the race to achieve artificial general intelligence (AGI), has led to large advancements in the cost-effectiveness, intelligence, and context lengths of LLMs. For instance, OpenAI's GPT-3.5 launched just a year ago, has already been surpassed in all aspects by newer models like GPT-4o and its variants. GPT-3.5 originally had a context length of 8,192, and we are already at 128k with GPT-4 and 1.5 million with Gemini (Google, 2024). With each release, the LLMs are also

getting better at their intelligence metrics. For example, when Anthropic's Claude 3.5 (www.anthropic.com, n.d.) was released it surpassed all other models and became the frontier model for a short time only to be taken over by the open-source LLM by meta, LLAMA-3.1. (Meta, 2024)

	Claude 3.5 Sonnet	Claude 3 Opus	GPT-4o	Gemini 1.5 Pro
Graduate level reasoning *GPQA, Diamond*	59.4%* 0-shot CoT	50.4% 0-shot CoT	53.6% 0-shot CoT	–
Undergraduate level knowledge *MMLU*	88.7%** 5-shot 88.3% 0-shot CoT	86.8% 5-shot 85.7% 0-shot CoT	– 88.7% 0-shot CoT	85.9% 5-shot –
Code *HumanEval*	92.0% 0-shot	84.9% 0-shot	90.2% 0-shot	84.1% 0-shot
Multilingual math *MGSM*	91.6% 0-shot CoT	90.7% 0-shot CoT	90.5% 0-shot CoT	87.5% 8-shot
Reasoning over text *DROP, F1 score*	87.1 3-shot	83.1 3-shot	83.4 3-shot	74.9 Variable shots
Mixed evaluations *BIG-Bench-Hard*	93.1% 3-shot CoT	86.8% 3-shot CoT	–	89.2% 3-shot CoT
Math problem-solving *MATH*	71.1% 0-shot CoT	60.1% 0-shot CoT	76.6% 0-shot CoT	67.7% 4-shot
Grade school math *GSM8K*	96.4% 0-shot CoT	95.0% 0-shot CoT	–	90.8% 11-shot

* Claude 3.5 Sonnet scores 67.2% on 5-shot CoT GPQA with maj@32
** Claude 3.5 Sonnet scores 90.4% on MMLU with 5-shot CoT prompting

This project will focus on GPT-4o and GPT-4o-mini as the LLM backend due to their advanced capabilities and reasonable cost-performance ratio. OpenAI's APIs are also more straightforward to integrate compared to other releases and their upgrades have been more consistent, thus saving a lot of development efforts in that regard and the focus can be put back on their actual integration into the game. However, it's worth noting that the LLM landscape is rapidly evolving, with models becoming cheaper, faster, and smarter at an impressive rate. So the final project might use an entirely different set of LLMs to ensure the quality of the narration response. Irrespective of the LLM used the goal is to keep the wrapper service consistent to the game so that even with a change in the backend API the overall system shouldn't drastically change.

2.1.1 Vision API:

While the project primarily focuses on text-based interactions, the integration of vision capabilities could significantly enhance the game experience. Current state-of-the-art vision APIs, such as GPT-4 Vision or Google's Gemini Vision offers powerful image understanding capabilities that could be leveraged in our game. This is particularly important in games because the NPCs and the narrative engine have to constantly see the visual changes that go on in the game. Unfortunately with the current state of technology, the vision APIs are very expensive and the delays are unreasonable for a real-time game setting, hence the project would not use the vision capabilities of the large language models, instead, the focus will be on visualizing the game progress through other means like data and values so that the LLM agent can still understand the level changes without having to look at the visual data..

2.2 Large Language Models in Games:

The integration of LLMs into game development has been an area of growing interest for both researchers and developers. Various studies have explored different aspects of LLM integration in gaming contexts:

1. Players and LLMs working together to solve quests (Rao et al., 2024), the paper explores how the LLM-NPCs can help the player to make the game more immersive and also help them in solving the game faster.
2. Dialogue Generation: Kalbiyev (2022) investigated the effectiveness of LLM-generated dialogues by comparing them to actual human-generated dialogue datasets from Fallout 4. This was 2022 so the project built upon fine-tuned GPT-2.
3. Interactive Narrative Scene Generation: (Kumaran et al., 2023) The paper mainly introduces a framework where the LLMs take care of the branching game narration while keeping the overall game direction in check.
4. The Turing Quest: (Chen Gao and Emami, 2023) The paper researches NPC script generation with the help of LLMs and integrates them into the game. The current project also explores

this with the perspective of even modern generation LLMs and also how the scripts can affect the overall narration.

These advancements lay the groundwork for the current project, which aims to push the boundaries of LLM integration in game development even further.

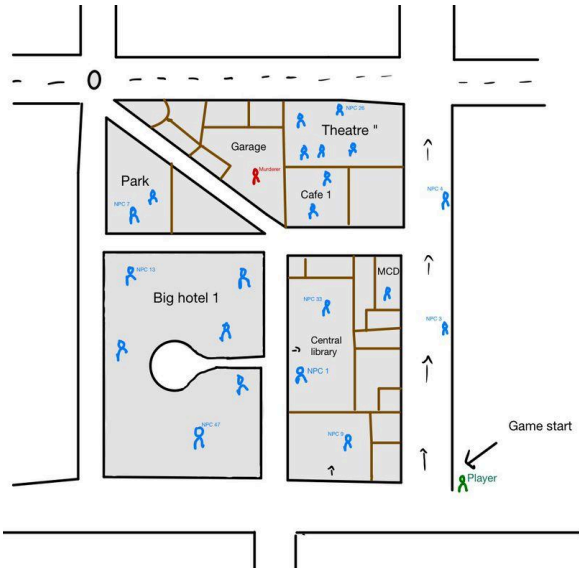
3. Game Implementation

3.1 Aims and Objectives

The primary aim of this project is to develop a proof-of-concept game that showcases the potential of LLM integration for dynamic narrative and gameplay experiences. ***The "dynamic narration" in the current context is the overall gameplay the player experiences while playing the game, which includes the mission, the ending of the game, the path to complete the mission, the NPC dialogues, the type of NPCs, etc. Things that will not be impacted by the dynamic narration are the level design, game start point, player look and feel, the main purpose of the player in the game, and the boundaries and rules of the game world.*** It is a detective-style adventure set in an urban environment, where player choices and interactions with NPCs directly influence the story's progression and outcome.

Key objectives include:

1. Implement a dialogue system driven by LLMs, allowing for natural, context-aware conversations between the player and NPCs.
2. Create a dynamic game flow where player decisions and actions have meaningful consequences on the narrative.
3. Develop a system for mass AI/NPC spawning with individualized behaviors and knowledge bases integrated with LLM agents per NPC instance.
4. Integrate LLM-driven decision-making into Unreal Engine's AI behavior trees.
5. Demonstrate the potential for replayability through a dynamically generated quest or its story and evolving NPC relation meaningful consequences on the narrative.



3.2 Game Design and Mechanics

The game world is a small area of a larger city with different Points-of-Interests (POIs) like the cafe, library, park and etc. It will also have a lot of NPCs walking around, there will be moving dummy vehicles that cannot be integrated with. The Player moves around the game by walking and the objective of the player is to find the murderer who has recently been involved with a case that the player is investigating. When the game starts player will have an initial hint about where the last time the eyewitness has seen the killer and a brief description of their appearance. From here the player traverses throughout the level to speak to different NPCs in different walks of life and the rest unfolds based on the decisions and conversations the player intends to make.

The NPCs are of three kinds, all unique LLMs Agent instance, being neutral, helpful, and with hidden motives to save the murderer. It depends on the player to detect them by talking to the NPCs or the other NPCs around them.

Screengrab of the Player interaction with NPC in the project

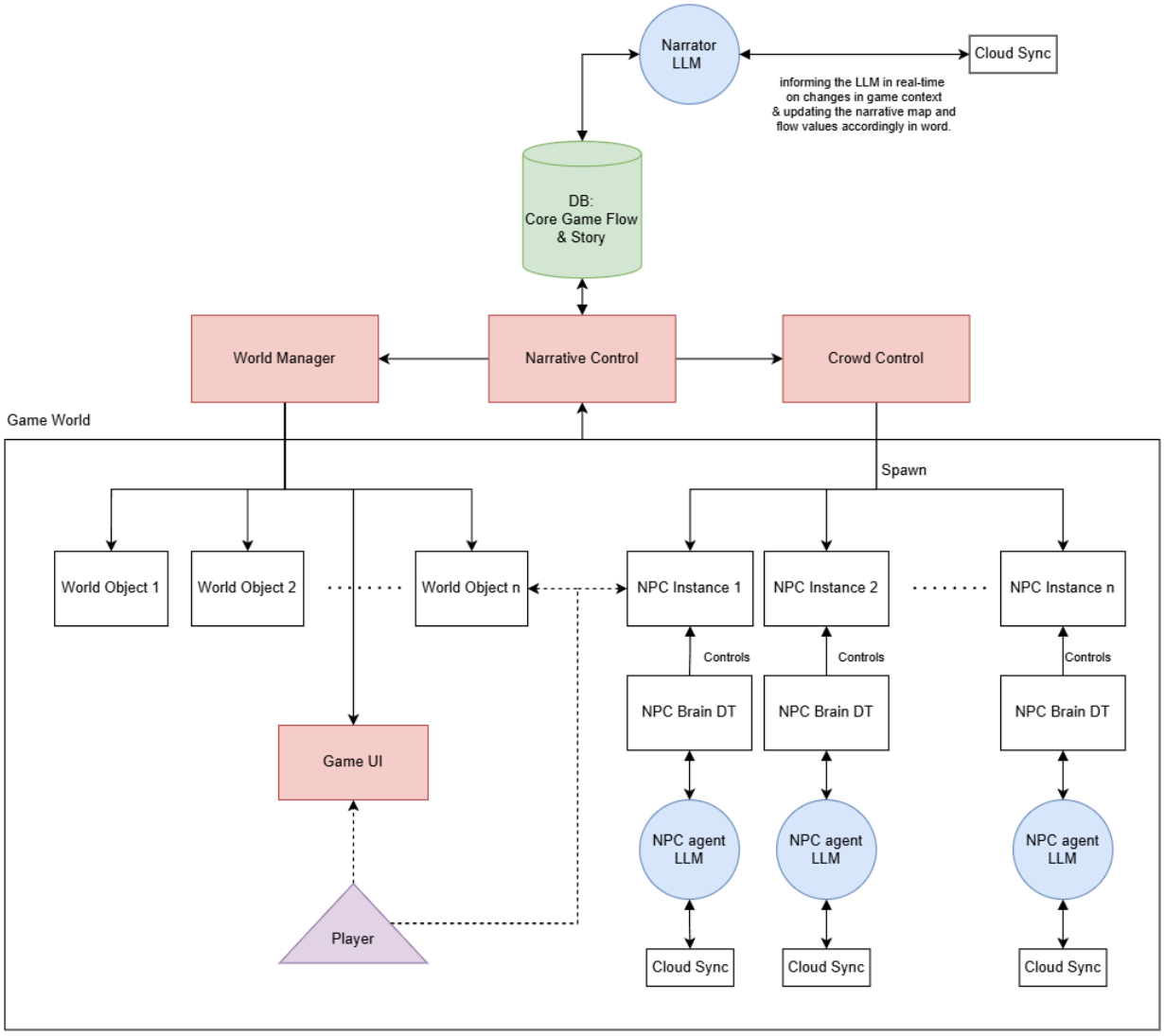
The game ends when either player finds the murderer successfully or the murderer realizes the player is looking for them and escapes from the city.

Players will always get potential questions generated by LLM to ask the NPCs and also an extra type box to type anything random to any NPC in the game, Whereas the NPCs in return will either respond both voice and text or text only. The conversation can go for lengths but at times the LLM agent can also decide to drop off from the conversation if the player appears to be wasting their time or tries to diverge from actual mission questions etc.

This adaptive system creates a highly replayable experience, as the narrative can unfold differently based on the player's choices and the LLM's dynamic content generation. The game state evolves constantly, with each interaction potentially affecting future dialogues, NPC behaviors, and story developments. By leveraging LLMs for all dialogue and narrative elements, the project has a game world that feels responsive and alive, where player choices have meaningful and often unpredictable consequences.

The world however will also have some preset rules so that the LLMs can freely interact and also to make sure the overall game never falls under a loop or becomes impossible to complete. Some rules are related to NPC spawning and their context timeouts to make sure the player can revisit the NPC that he or she previously spoke to and ask about some info they have attained. There also have to be enough NPCs around always in the game to make sure players get enough opportunities to interact with them. The game also poses a narrative engine that takes care of more gameplay rules to make sure the game ends and starts, the conversation does not diverge too far from the game’s core purpose, and the game hasn’t been forever going without an outcome, etc.

It was also key during the development of the project to make sure the game wasn’t making unnecessary API requests to keep the overall pricing in check.



Game and LLM Integration architecture

3.3 Technical Implementation

3.3.1 Development Environment

- Game Engine: Unreal Engine 5.3.2
- IDE: Rider 2024.1
- Version Control: GitHub
- API Testing: Postman
- LLM provider: OpenAI
- External:
    - MetaHumans by Epic
    - City Sample Crowds by Epic

3.3.2 LLM Integration

We'll be using a combination of OpenAI's GPT-4o API and GPT-4o-mini API for different aspects of the game:

- GPT-4o: Primary narration and dialogue generation and game flow management
- GPT-4o-mini: Secondary tasks such as environmental descriptions and minor NPC behaviors, reducing lag and cost.

API calls will also be optimized in general to balance response time and cost, with careful management of token usage.

3.3.3 Audio Implementation

The project is still in the works of implementing Text-To-Speech APIs of OpenAI to give voices to the agents and it will not implement Whisper for allowing players to directly speak to the Agents through their mic.

3.3.4 Unreal Engine Integration

The project will leverage several Unreal Engine plugins to streamline development. So far it has integrated:

- MetaHumans by Epic Games: For rapid creation of diverse NPC appearances
- Dialogue Plugin by CodeSpartan: To manage conversation flows and UI. This however was non-dynamic out of the box and had to be changed to make them dynamic.
- City Sample Crowds by Epic Games: For populating the game world with background NPCs

Custom C++ scripts will be developed to handle the integration between Unreal Engine's systems and the LLM APIs.

3.4 Pricing and Resource Management:

Pricing is always a big factor when scaling games like these to the consumers. So it will be one of the main aspects of consideration in this project to understand LLM cost. Here is the current breakdown: (This is subject to change as newer LLMs emerge)

3.4.1 LLM API Pricing:

GPT-4o API:

$5.00 / 1M input tokens + $15.00 / 1M output tokens

GPT-4o-mini API:

$0.150 / 1M input tokens + $0.600 / 1M output tokens

Whisper API:

$0.006 / minute (rounded to the nearest second)

Text-to-speech (TTS) API:

$15.00 / 1M characters

3.4.2 Other Pricings:

Unreal Engine, Rider, Github Copilot, GitHub Version Control are all currently free for students.

4. Workplan

4.1 Phase 1: Research and Design (Completed)

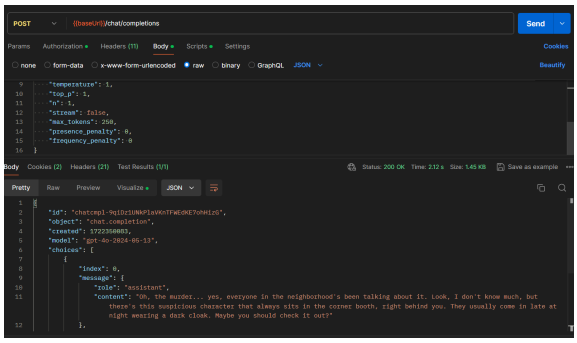
- Literature review and technology assessment
- LLM selection and initial API testing
- Unreal Engine project setup and plugin integration

4.2 Phase 2: Core Systems Development (July - Mid August) (Work in progress)

- Implement basic city environment in Unreal Engine ✔
- Develop LLM-driven dialogue system (WIP)
- Create NPC behavior framework ✔

4.3 Phase 3: Gameplay and Narrative Integration (August)

- Implement investigation mechanics
- Set up level Points-Of-Interests, and unique NPC spawning
- Create quest and clue distribution algorithms and prompt engineering
- Refine NPC interactions and behaviors



4.4 Phase 4: Testing and Refinement (September to January)

- Conduct playtests and gather feedback
- Optimize LLM prompts and responses, and Lag
- Text streaming for API response.
- Balance gameplay difficulty and narrative coherence
- Fine-tune performance and reduce API costs
- If time permits integrate TTS and Whisper.

4.5 Phase 5: Documentation and Presentation (January)

- Compile final project report
- Prepare demonstration video
- Create project presentation materials

Throughout the development process, we'll use agile methodologies, conducting weekly sprints and regular progress reviews to ensure we stay on track and can adapt to any challenges that arise.

5. Ethics

OpenAI Usage Policies (OpenAI) require the project to provide clear disclosure to end users that the TTS voice they are hearing is AI-generated and not a human voice. AI-generated voices should not be used to promote harmful activities, mislead users, or engage in illegal activities. Avoid generating sexually explicit or inappropriate content for minors, unless it's for educational or scientific purposes within the game context. Avoid misrepresentation: Don't use the AI voices to impersonate real individuals or organizations without consent.

Unreal Engine licensing (Epic Games) is free for students. The free license applies to educational use without revenue limits, else the threshold is at $1 million USD in gross revenue. Release notification: If the developer eventually releases the project commercially, They need to submit a Release Notification Form. (here)

6. Conclusions

The project's progress so far has hinted at the potential use case of current generation LLMs in games, however, there are still some concerns about the context length, pricing, and API response times of the LLM to make a concept like this truly feasible in commercial games. The resources further will be used to make sure the project works on such issues and also to make sure the game is immersive and dynamic and the conversations feel real within the context.

It was also observed during the development that the narration can truly be unique, including the overall scope itself, for example in the current project the overall scope is "investigation". This also highlights how dynamic games of the future can be and how much the LLMs play a part in it. Although these extended aspects won't be explored in the current project as they need future vision APIs, dynamic animations, etc, it will however prove the potential of it all by the end of the completion and I think that is the point of it.

7. References

PlayStation. (n.d.). Discover UNCHARTED. [online] Available at: https://www.playstation.com/en-gb/uncharted/.

Games, R. (n.d.). Red Dead Redemption. [online] Rockstar Games. Available at: https://www.rockstargames.com/reddeadredemption.

Rao, S., Xu, W., Xu, M., Leandro, J., Lobb, K., DesGarennes, G., Brockett, C. and Dolan, B. (2024). Collaborative Quest Completion with LLM-driven Non-Player Characters in Minecraft. [online] arXiv.org. Doi: https://doi.org/10.48550/arXiv.2407.03460

Kalbiyev, A. (2022). Affective dialogue generation for video games. [online] essay.utwente.nl. Available at: https://essay.utwente.nl/89325/.

Google. (2024). Gemini breaks new ground with a faster model, longer context, AI agents, and more. [online] Available at: https://blog.google/technology/ai/google-gemini-update-flash-ai-assistant-io-2024 [Accessed 30 Jul. 2024].

www.anthropic.com. (n.d.). Introducing Claude 3.5 Sonnet. [online] Available at: https://www.anthropic.com/news/claude-3-5-sonnet.

Meta. (2024). The Llama 3 Herd of Models | Research - AI at Meta. [online] Available at: https://ai.meta.com/research/publications/the-llama-3-herd-of-models/.

Kumaran, V., Rowe, J., Mott, B. and Lester, J. (2023). SceneCraft: Automating Interactive Narrative Scene Generation in Digital Games with Large Language Models. Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, [online] 19(1), pp.86–96. Doi: https://doi.org/10.1609/aiide.v19i1.27504.

Chen Gao, Q. and Emami, A. (2023). The Turing Quest: Can Transformers Make Good NPCs? [online] pp.93–103. Available at: https://aclanthology.org/2023.acl-srw.17.pdf

Usage Policies by OpenAI: https://openai.com/policies/usage-policies/

Unreal Engine. (n.d.). Unreal Engine (UE5) licensing options. [online] Available at: https://www.unrealengine.com/en-US/license.

8. Appendix

Important links to the LLMs that were discussed in the report:
Claude 3.5 Sonnet:
Introducing Claude 3.5 Sonnet \ Anthropic
Llama 3 to download:
Llama 3.1 - a meta-llama Collection
Llama 3 paper:
The Llama 3 Herd of Models | Research - AI at Meta
GPT4o and GPT4o-mini:
API Reference | OpenAI Platform

Table Of Contents: