**A picture containing text

Description automatically generated**

**Department of Informatics**

**University of Leicester**

**CO7201 Individual Project**

**Preliminary Report**

**Generating Game Narratives using an**

**AI Language Model**

**Rohan Anand**

[**ra470@student.le.ac.uk**](mailto:ra470@student.le.ac.uk)

**ra470**

**Project Supervisor: Victoria R. Wright**

**Second Marker: Dr. Fabricio Goes**

**Word Count: [1500]**

**30/06/2023**

**DECLARATION**

All sentences or passages quoted in this report, or computer code of any form whatsoever used and/or submitted at any stages, which are taken from other people’s work have been specifically acknowledged by clear citation of the source, specifying author, work, date and page(s). Any part of my own written work, or software coding, which is substantially based upon other people’s work, is duly accompanied by clear citation of the source, specifying author, work, date and page(s). I understand that failure to do this amounts to plagiarism and will be considered grounds for failure in this module and the degree examination as a whole.

Name: Rohan Anand

Date: 30-06-2023

Contents

[**1.** **Aims and Objectives** 3](#_Toc65510906)

[**2.** **Requirements** 4](#_Toc65510907)

[**3.** **Technical Specification** 4](#_Toc65510908)

[**4.** **Requirements Evaluation Plan** 6](#_Toc65510909)

[**5.** **Background Research and Reading list** 6](#_Toc65510910)

[**6.** **Time-plan and Risk Plan** 7](#_Toc65510911)

[**7.** **References** 8](#_Toc65510912)

# **Aims and Objectives**

**Introduction:**

The aim of this preliminary project report is to outline the objectives and motivations behind the development of a Game Narrative Generator utilizing an AI language model. This report will provide an overview of the project's relevance and significance, as well as discuss the main challenges that need to be addressed during its implementation.

**Significance of Game Narratives:** Game narratives play a crucial role in providing players with immersive and engaging experiences. A well-crafted game narrative can enhance the player's sense of immersion, emotional investment, and overall enjoyment. However, creating compelling and diverse narratives can be a challenging task for game developers, as it requires substantial time, effort, and creative expertise.

**Role of AI-Language Models:** AI language models have shown tremendous potential in various creative domains, including natural language generation and storytelling. These models can generate coherent and contextually relevant text, making them ideal tools for automating the process of game narrative generation. By leveraging the capabilities of AI language models, game developers can streamline the narrative creation process, leading to more dynamic and interactive gaming experiences.

**Project Relevance:**

The development of a web application as a Game Narrative Generator using a chosen AI language model holds significant relevance for several reasons. Firstly, it offers a novel approach to addressing the challenges faced by game developers in creating captivating narratives. By automating the narrative generation process, developers can save time and resources, allowing them to focus on other critical aspects of game development.

Secondly, this project provides an opportunity to explore the potential of AI language models in the context of interactive storytelling and branched narratives. Traditional linear narratives can be limiting in terms of player agency and immersion. By employing AI language models, the web application can generate branching narratives that respond to player choices, creating a more personalized and engaging gameplay experience.

Lastly, this project serves as a learning opportunity to gain a comprehensive understanding of AI language models, their applications, and their limitations. By conducting research and evaluating the capabilities and limitations of the chosen AI language model, valuable insights can be gained, contributing to the broader field of AI-driven creative endeavours.

**Challenges:**

The use of AI-generated narratives in video games is not without its challenges. One of the primary concerns is ensuring that the content generated by LLMs (Large Language Models) remains coherent and consistent with the overall narrative structure of the game. To address this issue, game developers must carefully design the algorithms and parameters that guide the AI’s narrative generation process, ensuring that the resulting content is both engaging and thematically appropriate.

# **Requirements**

**Essential:**

1. Web Application: Develop a Game Narrative Generator using the chosen AI language model, GPT4All.
2. User Interface: Create a user-friendly interface allowing users to navigate through game world selection, time period selection, character design, and customization options.
3. Narrative Generation: Implement the AI language model to generate narratives based on user inputs and selected settings.
4. Branched Narratives: Provide the user with multiple branching narrative options to continue the story journey.
5. Story Conclusion: Allow the user to choose when to conclude the story journey, and generate an appropriate ending based on their decision.
6. Story Scripting: Provide an option for users to view and script the entire story they have created so far.

**Recommended:**

1. Limitations Evaluation: Conduct an analysis of test case fail rates to understand the limitations and shortcomings of the chosen AI language model.
2. System Limitations: Evaluate the system limitations and identify areas for improvement or additional features to enhance the overall performance.

**Optional:**

1. Exploratory Study: Conduct research and perform an exploratory study on how AI language models generate interactive narratives and examine the impact of generative agents powered by AI on creating believable human behaviour simulacrum in games.
2. Performance Enhancement: Propose solutions to enhance the language model's performance by further researching areas such as Creative Computing, Computational Creativity, and Artificial Intelligence.

# **Technical Specification**

**AI Language Model:**

For our use case of a game narrative generator, we have chosen GPT-4ALL as our preferred AI language model. While both ChatGPT4 and GPT-4ALL are powerful LLMs, GPT-4ALL excels in generating creative and original stories, which is essential for crafting engaging game narratives. It has been trained on a diverse dataset, including books, articles, and other forms of creative writing, enabling it to produce unique and compelling storylines. By leveraging GPT-4ALL, we can provide players with immersive and captivating game narratives that are filled with imaginative elements, enhancing the overall gaming experience.

**Backend Development:**

a. Language: Python will be used as the primary programming language for backend development.

b. Framework: The web application will be developed using the Python Flask framework. Flask is a lightweight and flexible framework that enables rapid development of web applications.

c. LangChain: LangChain, a powerful tool for working with Large Language Models (LLMs), will be utilized. LangChain provides an interface to interact with the GPT4All model and handle the complexities of working with large language models efficiently.

**Flask and LangChain** together in the web application: Flask provide a framework for building web applications in Python, while LangChain allows you to integrate large language models like GPT4All. By combining Flask and LangChain, you can create a dynamic web application that leverages the power of language models for generating text-based responses and interactions.

**Frontend Development:**

a. HTML, CSS, and JavaScript will be used for front-end development.

b. User Interface (UI): Design an intuitive and user-friendly UI to allow users to navigate through game world selection, time period selection, character design, and customization options. The UI should also display generated narratives and provide options for branching and concluding the story.

**API Integration:**

The GPT4All Python API allows you to work with GPT4All language models. You can initialize the API, download and retrieve models, generate responses based on input prompts, and interact with the models through chat-like conversations. The API provides methods for model initialization, downloading models, generating responses, and listing available models.

**Testing and Debugging:**

a. Implement unit testing and integration testing to ensure the correctness and reliability of the application.

b. Debug and handle exceptions to provide a robust and error-free user experience.

**Documentation:**

a. Create comprehensive documentation, including installation instructions, set up guidelines, and code documentation, to facilitate project maintenance and future development.

b. Document the APIs, including their endpoints, input parameters, and expected responses, for easy integration with frontend or third-party applications.

**Security:**

a. Implement necessary security measures, such as input validation and output sanitization, to protect against potential vulnerabilities and malicious attacks.

b. Follow secure coding practices to ensure the application is resilient to common security threats.

**Scalability:** Design the application with scalability in mind to accommodate potential future growth in user base and increased demand.

# **Requirements Evaluation Plan**

To evaluate the system and verify its effectiveness, the following criteria and testing methods will be employed:

* **System Performance:**

Measure response time to ensure the timely generation of narratives.

* **Narrative Quality:**

1. Assess the coherence and contextual relevance of generated narratives.
2. Evaluate creativity and originality to avoid repetition and predictability.

* **User Interaction:**

1. Test branching narratives to provide engaging story options.
2. Assess user customization for personalized game narratives.

* **Testing Approach:**

1. Conduct unit testing to verify component functionality.
2. Perform integration testing to ensure seamless system communication.
3. Engage users, including the developer and users, for user testing and feedback.

* **Evaluation Participants:**

1. Developer: Assess system adherence to requirements and objectives.
2. External Evaluators: Involve the supervisor, second marker, or if possible, game developers, testers, and potential end-users for unbiased feedback.

* **Evaluation Process:**

1. Conduct trial runs with varied input scenarios and interactions.
2. Observe system behaviour, noting issues like slow response time or repetitive narratives.
3. Analyse findings, collect user feedback, and make necessary improvements.

By following this evaluation plan, the system's performance, narrative quality, user interaction, and overall software quality will be assessed to ensure the project achieves its goals of providing a reliable and engaging game narrative generator.

# **Background Research and Reading list**

* GPT-4ALL Documentation and Resources:

GPT-4ALL GitHub Repository:

<https://github.com/nomic-ai/gpt4all/tree/main/gpt4all-bindings/python>

* OpenAI API Documentation: <https://platform.openai.com/docs/api-reference/introduction>
* Getting Started with LangChain Python: [https://python.LangChain.com/docs/get\_started/introduction](https://python.langchain.com/docs/get_started/introduction)
* APIs in the LangChain Python package: [https://api.python.LangChain.com/en/latest/](https://api.python.langchain.com/en/latest/)

Flask Web Framework:

* Flask Documentation: <https://flask.palletsprojects.com/>
* Flask Mega-Tutorial by Miguel Grinberg: <https://blog.miguelgrinberg.com/post/the-flask-mega-tutorial-part-i-hello-world>

Web Development and HTML/CSS:

* Mozilla Developer Network (MDN) Web Docs: <https://developer.mozilla.org/>
* W3Schools: <https://www.w3schools.com/>

User Experience (UX) and User Interface (UI) Design:

* Interaction Design Foundation:<https://www.interaction-design.org/>
* NNGroup - Usability and UX Research:<https://www.nngroup.com/>

AI and Natural Language Processing (NLP) Concepts:

* Stanford NLP Group:<https://nlp.stanford.edu/>
* TensorFlow NLP Tutorials:<https://www.tensorflow.org/text/tutorials>

# **Time-plan and Risk Plan**

**Time Plan:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Task** | **Start Date** | **End Date** | **Duration** |
| Research and Background Reading | 09/06/2023 | 16/06/2023 | 1 week |
| Gather Requirements | 17/06/2023 | 23/06/2023 | 1 week |
| Design User Interface | 24/06/2023 | 30/06/2023 | 1 week |
| Develop Backend Functionality | 01/07/2023 | 14/07/2023 | 2 weeks |
| Integrate GPT-4ALL | 15/07/2023 | 28/07/2023 | 2 weeks |
| Implement Branched Narratives | 29/07/2023 | 04/08/2023 | 1 week |
| User Testing and Feedback | 05/08/2023 | 11/08/2023 | 1 week |
| Refinement and Bug Fixing | 12/08/2023 | 18/08/2023 | 1 week |
| Documentation and Finalization | 19/08/2023 | 25/08/2023 | 1 week |
| Deployment and Launch | 26/08/2023 | 08/09/2023 | 2 weeks |

**Risk Plan:**

As the sole developer with guidance from my project supervisor, proactive risk management is crucial. The risk plan includes:

1. Risk Identification: Identify potential risks and uncertainties in technical, operational, and external aspects.
2. Risk Assessment: Evaluate the likelihood and impact of each risk to prioritize and address them effectively.
3. Risk Mitigation: Develop contingency plans, and preventive measures, and allocate resources to mitigate risks.
4. Regular Review and Update: Continuously review and update the risk plan to adapt to evolving circumstances and emerging risks.

This risk plan minimizes negative impacts on the project. Regular communication with my supervisor ensures effective risk management. Its implementation ensures successful project completion within the designated timeframe.

# **References**

1. "Natural Language Processing with Python" by Steven Bird, Ewan Klein, and Edward Loper
2. Deep Learning with Python" by François Chollet (Chapter on Generative Models and GANs)
3. "The Creativity Code: Art and Innovation in the Age of Artificial Intelligence" by Marcus du Sautoy
4. "Flask Web Development with Python Tutorial" by Corey Schafer (Online tutorial series)
5. "Flask Web Development: Developing Web Applications with Python" by Miguel Grinberg
6. "The Elements of User Experience: User-Centered Design for the Web" by Jesse James Garrett
7. "The Art of Game Design: A Book of Lenses" by Jesse Schell