

# Geração de Mapas para Jogos Roguelike a Partir de Descrições Textuais Utilizando Modelos de Larga Escala

Gustavo Gurgel (Orientador: Cristiano Bacelar)

[gusgurgel@alu.ufc.br](mailto:gusgurgel@alu.ufc.br)

**Universidade Federal do Ceará**

21 de janeiro de 2026

# Sumário

- 1 Introdução
- 2 Fundamentação
- 3 Trabalhos Relacionados

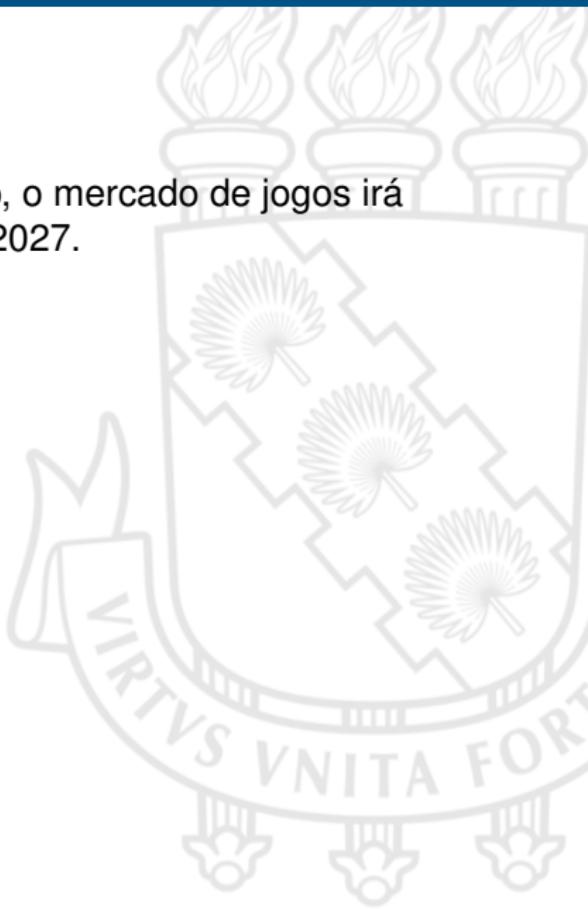
- 4 Metodologia
- 5 Resultados
- 6 Conclusão



UNIVERSIDADE  
FEDERAL DO CEARÁ

# Escopo Geral do Problema

- Segundo levantamentos da Newzoo, o mercado de jogos irá arrecadar **\$198 bilhões** no ano de 2027.



# Escopo Geral do Problema

- Segundo levantamentos da Newzoo, o mercado de jogos irá arrecadar **\$198 bilhões** no ano de 2027.
- Nesse mercado, os *Roguelikes*, um gênero muito antigo (1980), continua recebendo alto reconhecimento. Dentro jogos desse gênero temos:

# Escopo Geral do Problema

- Segundo levantamentos da Newzoo, o mercado de jogos irá arrecadar **\$198 bilhões** no ano de 2027.
- Nesse mercado, os *Roguelikes*, um gênero muito antigo (1980), continua recebendo alto reconhecimento. Dentro jogos desse gênero temos:
  - **Balatro (2024)**: Concorreu como um dos 6 melhores jogos de 2024.

# Escopo Geral do Problema

- Segundo levantamentos da Newzoo, o mercado de jogos irá arrecadar **\$198 bilhões** no ano de 2027.
- Nesse mercado, os *Roguelikes*, um gênero muito antigo (1980), continua recebendo alto reconhecimento. Dentro jogos desse gênero temos:
  - **Balatro (2024)**: Concorreu como um dos 6 melhores jogos de 2024.
  - **Hell Clock (2025)**: Roguelike brasileiro inspirado na Guerra dos Canudos

# Escopo Geral do Problema

- Segundo levantamentos da Newzoo, o mercado de jogos irá arrecadar **\$198 bilhões** no ano de 2027.
- Nesse mercado, os *Roguelikes*, um gênero muito antigo (1980), continua recebendo alto reconhecimento. Dentro jogos desse gênero temos:
  - **Balatro (2024)**: Concorreu como um dos 6 melhores jogos de 2024.
  - **Hell Clock (2025)**: Roguelike brasileiro inspirado na Guerra dos Canudos
- Um característica predominate nos *Roguelikes* é a geração de conteúdos (items, missões, inimigos) através de algoritmos.

# Escopo Geral do Problema

- Segundo levantamentos da Newzoo, o mercado de jogos irá arrecadar **\$198 bilhões** no ano de 2027.
- Nesse mercado, os *Roguelikes*, um gênero muito antigo (1980), continua recebendo alto reconhecimento. Dentro jogos desse gênero temos:
  - **Balatro (2024)**: Concorreu como um dos 6 melhores jogos de 2024.
  - **Hell Clock (2025)**: Roguelike brasileiro inspirado na Guerra dos Canudos
- Um característica predominate nos *Roguelikes* é a geração de conteúdos (items, missões, inimigos) através de algoritmos.
- **Problema:** Em geral, a geração desses conteúdos é guiada através de **parâmetros numéricos, muitas vezes de difícil interpretação**.



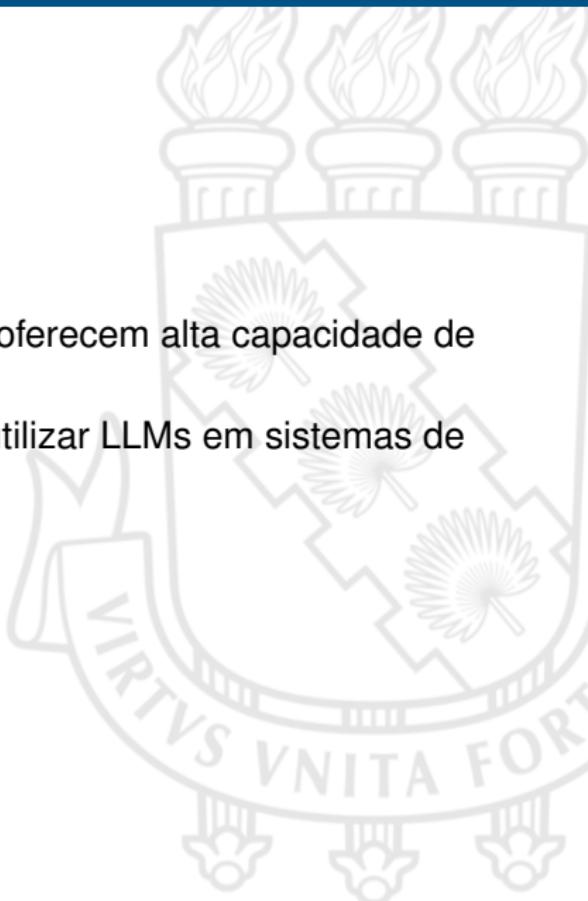
# Solução: PCG + LLMs

- Os grandes modelos de linguagem oferecem alta capacidade de interpretação de linguagem natural.



# Solução: PCG + LLMs

- Os grandes modelos de linguagem oferecem alta capacidade de interpretação de linguagem natural.
- Assim, destaca-se a oportunidade utilizar LLMs em sistemas de geração de conteúdo para jogos.

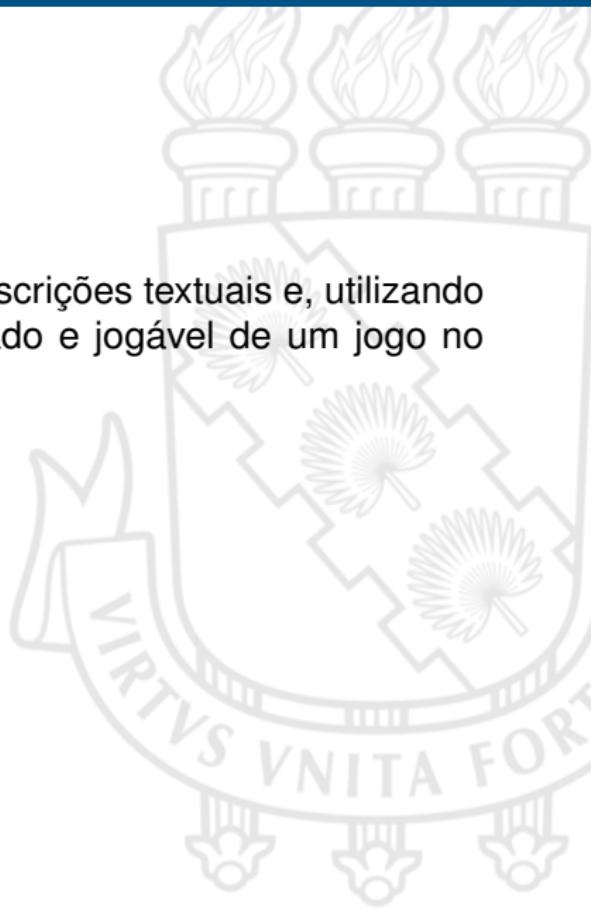


# Solução: PCG + LLMs

- Os grandes modelos de linguagem oferecem alta capacidade de interpretação de linguagem natural.
- Assim, destaca-se a oportunidade utilizar LLMs em sistemas de geração de conteúdo para jogos.
- Em vez de inserir valores numéricos, o utilizador descreve o conteúdo a ser gerado utilizando linguagem natural.

# Objetivos

Desenvolver um sistema que recebe descrições textuais e, utilizando um LLM, traduz em um nível estruturado e jogável de um jogo no gênero *Roguelike*.



# Objetivos

Desenvolver um sistema que recebe descrições textuais e, utilizando um LLM, traduz em um nível estruturado e jogável de um jogo no gênero *Roguelike*.

- Desenvolver um pipeline de geração que receba uma descrição textual e produza uma representação de mapa estruturada.

# Objetivos

Desenvolver um sistema que recebe descrições textuais e, utilizando um LLM, traduz em um nível estruturado e jogável de um jogo no gênero *Roguelike*.

- Desenvolver um pipeline de geração que receba uma descrição textual e produza uma representação de mapa estruturada.
- Investigar e aplicar técnicas de engenharia de prompt para guiar o LLM de forma eficaz.

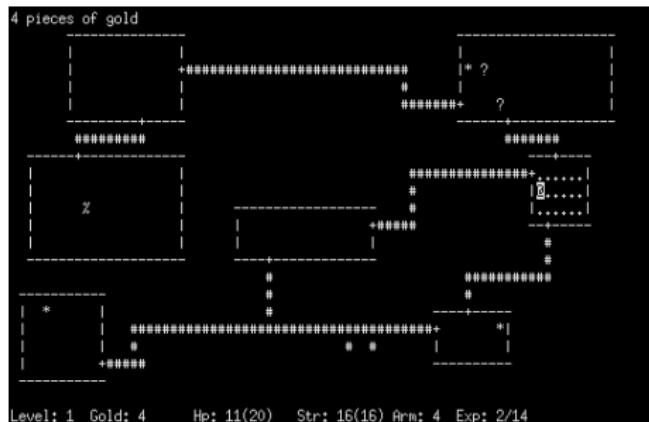
# Objetivos

Desenvolver um sistema que recebe descrições textuais e, utilizando um LLM, traduz em um nível estruturado e jogável de um jogo no gênero *Roguelike*.

- Desenvolver um pipeline de geração que receba uma descrição textual e produza uma representação de mapa estruturada.
- Investigar e aplicar técnicas de engenharia de prompt para guiar o LLM de forma eficaz.
- Implementar um protótipo capaz de processar os dados do mapa e renderizá-los visualmente.

# Origem dos *Roguelikes*

Figura: Rogue (1980)



- Em 1980 surge **Rogue**, um jogo de terminal onde o jogador explorar masmorras geradas aleatoriamente, enfrentando monstros, coletando itens e tentando chegar ao final em uma única vida.

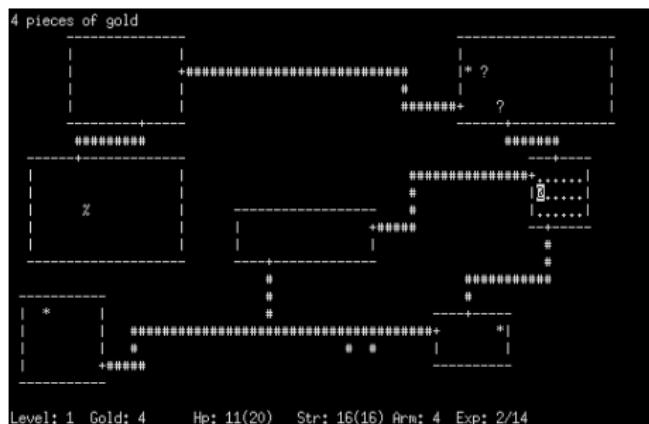
Fonte: Wikipedia



UNIVERSIDADE  
FEDERAL DO CEARÁ

# Origem dos *Roguelikes*

Figura: Rogue (1980)



- Em 1980 surge **Rogue**, um jogo de terminal onde o jogador explorar masmorras geradas aleatoriamente, enfrentando monstros, coletando itens e tentando chegar ao final em uma única vida.
- Esse jogo foi tão marcante que deu origem a um gênero inteiro chamado Roguelike, que se baseia em elementos como:

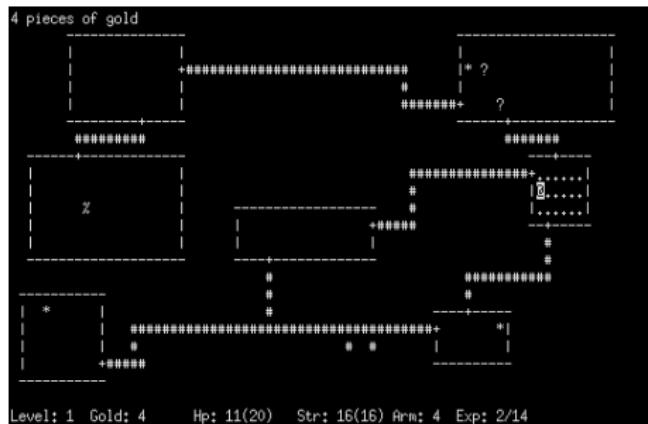
Fonte: Wikipedia



UNIVERSIDADE  
FEDERAL DO CEARÁ

# Origem dos *Roguelikes*

Figura: Rogue (1980)



- Em 1980 surge **Rogue**, um jogo de terminal onde o jogador explorar masmorras geradas aleatoriamente, enfrentando monstros, coletando itens e tentando chegar ao final em uma única vida.
- Esse jogo foi tão marcante que deu origem a um gênero inteiro chamado Roguelike, que se baseia em elementos como:
  - Geração procedural.

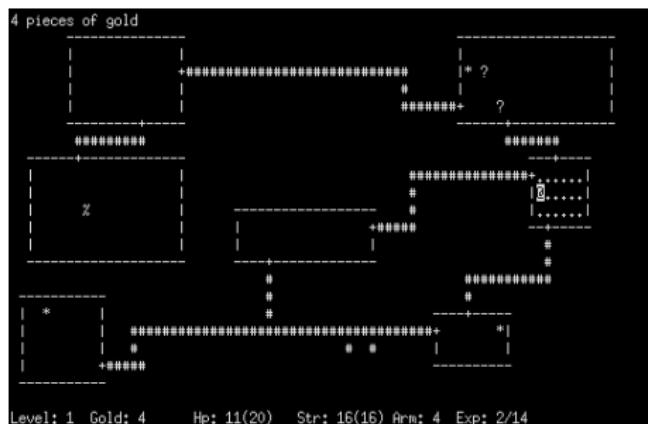
Fonte: Wikipedia



UNIVERSIDADE  
FEDERAL DO CEARÁ

# Origem dos *Roguelikes*

Figura: Rogue (1980)



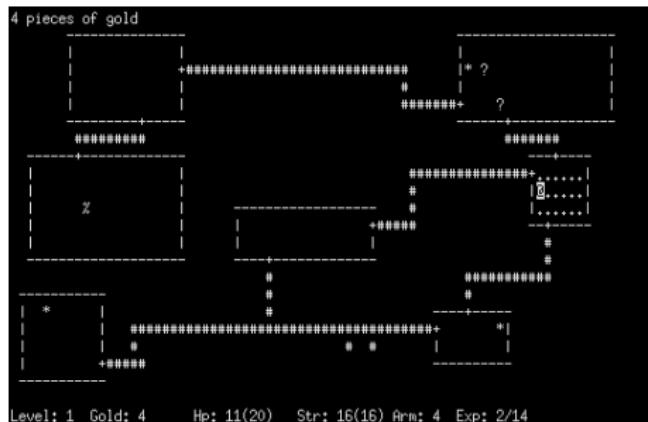
Fonte: Wikipedia

- Em 1980 surge **Rogue**, um jogo de terminal onde o jogador explorar masmorras geradas aleatoriamente, enfrentando monstros, coletando itens e tentando chegar ao final em uma única vida.
- Esse jogo foi tão marcante que deu origem a um gênero inteiro chamado Roguelike, que se baseia em elementos como:
  - Geração procedural.
  - Permadeath (morte permanente).



# Origem dos *Roguelikes*

Figura: Rogue (1980)



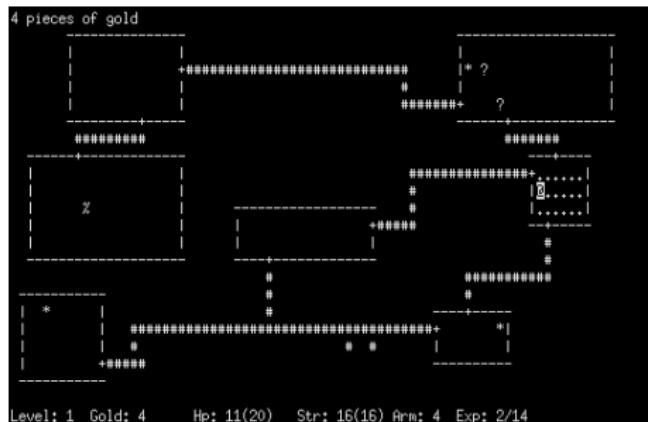
Fonte: Wikipedia

- Em 1980 surge **Rogue**, um jogo de terminal onde o jogador explorar masmorras geradas aleatoriamente, enfrentando monstros, coletando itens e tentando chegar ao final em uma única vida.
- Esse jogo foi tão marcante que deu origem a um gênero inteiro chamado Roguelike, que se baseia em elementos como:
  - Geração procedural.
  - Permadeath (morte permanente).
  - Exploração estratégica.



# Origem dos *Roguelikes*

Figura: Rogue (1980)



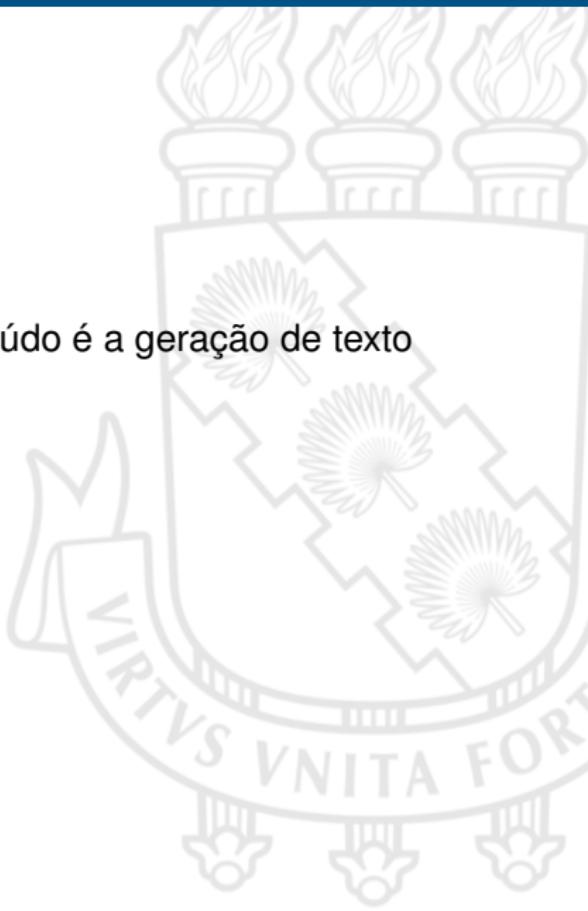
Fonte: Wikipedia

- Em 1980 surge **Rogue**, um jogo de terminal onde o jogador explorar masmorras geradas aleatoriamente, enfrentando monstros, coletando itens e tentando chegar ao final em uma única vida.
- Esse jogo foi tão marcante que deu origem a um gênero inteiro chamado Roguelike, que se baseia em elementos como:
  - Geração procedural.
  - Permadeath (morte permanente).
  - Exploração estratégica.
  - Alta rejogabilidade.



# Geração de texto com LLMs

- Outro exemplo de geração de conteúdo é a geração de texto utilizando LLMs.

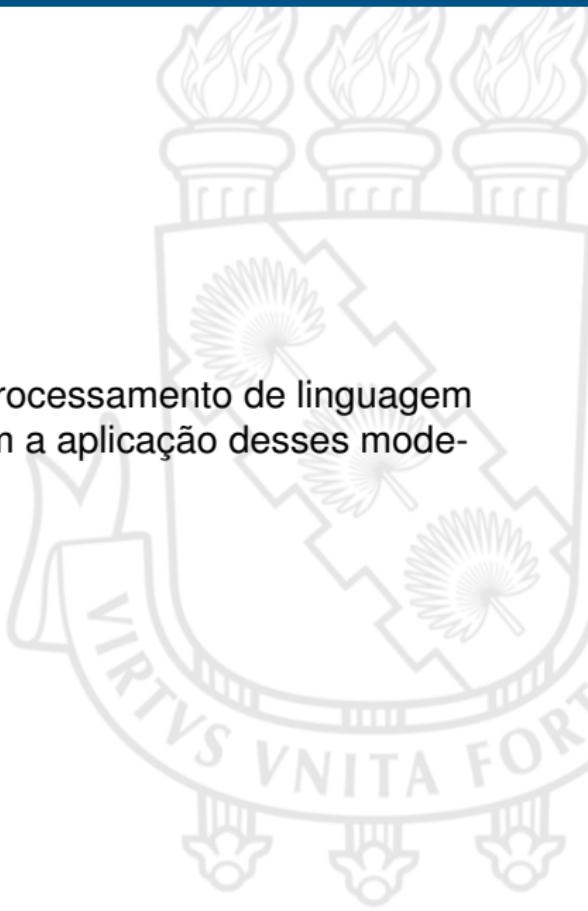


# Geração de texto com LLMs

- Outro exemplo de geração de conteúdo é a geração de texto utilizando LLMs.
- Fatores como definição da arquitetura *Transformer*, maior capacidade computacional e volumes gigantescos de dados textuais permitiram o surgimento de Modelos de Linguagem com capacidades notáveis no processamento de linguagem natural.

# Trabalhos Relacionados

Utilizando a proficiência dos LLMs no processamento de linguagem natural, os seguintes trabalhos exploram a aplicação desses modelos na geração de mapas:



# Trabalhos Relacionados

Figura: Trabalhos Relacionados

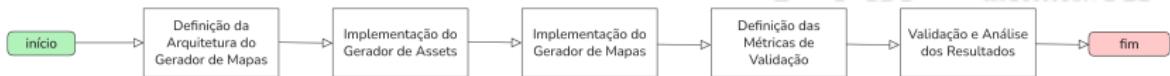
Trabalho	Geração de níveis a partir de descrições textuais	Foco em engenharia de prompt	Saída em formato estruturado (e.g., JSON)	Abordagem focada no gênero Roguelike
Word2World	x	x		
MarioGPT	x			
Auxtero	x		x	
LLMaker	x	x		
<b>Trabalho Proposto</b>	<b>x</b>	<b>x</b>	<b>x</b>	<b>x</b>

Fonte: Elaborado pelo autor



# Metodologia do Trabalho

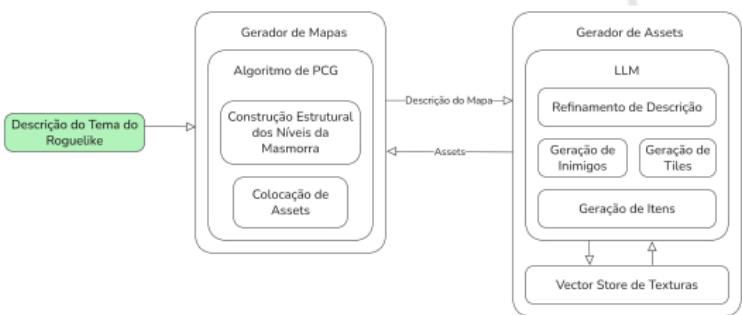
Figura: Fluxograma da Metodologia



Fonte: Elaborado pelo autor

# Definição da Arquitetura do Gerador de Mapas

Figura: Arquitetura Híbrida proposta



Fonte: Elaborado pelo autor

Conforme demonstrado por Yan et al., modelos de estado da arte, como o *GPT-4*, apresentam dificuldades substanciais em tarefas que exigem compreensão precisa de coordenadas, como plotar pontos em espaços 2D/3D ou executar algoritmos de busca de caminho.



# Implementação do Gerador de Assets

- **Problema 1:** Saídas não estruturadas dos LLMs não são processáveis pelo Gerador de Mapas.

# Implementação do Gerador de Assets

- **Problema 1:** Saídas não estruturadas dos LLMs não são processáveis pelo Gerador de Mapas.
- **Solução:** Utilização da técnica de *Structured Outputs*. Uso modelos ajustados para responder seguindo um esquema JSON.

# Implementação do Gerador de Assets

- **Problema 1:** Saídas não estruturadas dos LLMs não são processáveis pelo Gerador de Mapas.
- **Solução:** Utilização da técnica de *Structured Outputs*. Uso de modelos ajustados para responder seguindo um esquema JSON.
- **Problema 2:** O LLM gera saídas textuais, porém o jogo precisa de ativos visuais para representar itens, inimigos e ambientes.

# Implementação do Gerador de Assets

- **Problema 1:** Saídas não estruturadas dos LLMs não são processáveis pelo Gerador de Mapas.
- **Solução:** Utilização da técnica de *Structured Outputs*. Uso modelos ajustados para responder seguindo um esquema JSON.
  
- **Problema 2:** O LLM gera saídas textuais, porém o jogo precisa de ativos visuais para representar items, inimigos e ambientes.
- **Solução:** Utilização da técnica de *RAG*. Um banco vetorial guarda tuplas de descrições e images. O código pode utilizar saídas textuais do LLM para recuperar imagens.

# Implementação do Gerador de Assets

Figura: Funções de Query do Banco

Banco= items  
Query= "A greatsword"

Banco= entities  
Query= "A dwarf"

Banco= environments  
Query= "A stone wall"



Fonte: Elaborado pelo autor

Utilizando o pacote de tiles "Kenney 1-Bit Pack" foram feitos 3 bancos vetoriais: Entities (88 items), Items (183 items) e Environment (218 items)



UNIVERSIDADE  
FEDERAL DO CEARÁ

# Implementação do Gerador de Assets

Figura: Fluxograma de Geração de um Pacote de Assets



Fonte: Elaborado pelo autor

# Implementação do Gerador de Assets

Figura: Tecnologias Utilizadas no Gerador de Assets



LangChain



Chroma

Fonte: Elaborado pelo autor

# Implementação do Gerador de Mapas

**Entrada:** Restrições Geométricas ( $D_{mapa}$ ), Limite de Tentativas ( $N_{max}$ )

**Saída:** Topologia da Masmorra ( $Salas$ )

**Início**

```
    Salas ← ∅;  
    para  $i \leftarrow 0$  até  $N_{max}$  faça  
        Candidata ← GerarRetanguloAleatorio( $D_{mapa}$ );  
        se HalIntersecao(Candidata, Salas) então  
            | continuar;  
        fim  
        EscavarNoMapa(Candidata);  
        se Salas não está vazia então  
            | Alvo ← ObterUltimaSala(Salas);  
            | CriarCorredorConectando(Alvo, Candidata);  
        fim  
        Adicionar Candidata ao conjunto Salas;  
    fim  
    retorna Salas;  
fim
```

**Algorithm 1:** Geração Procedural da Estrutura da Masmorra



# Implementação do Gerador de Mapas

Figura: Tecnologias Utilizadas no Gerador de Jogos



Fonte: Elaborado pelo autor

# Definição das Métricas de Validação

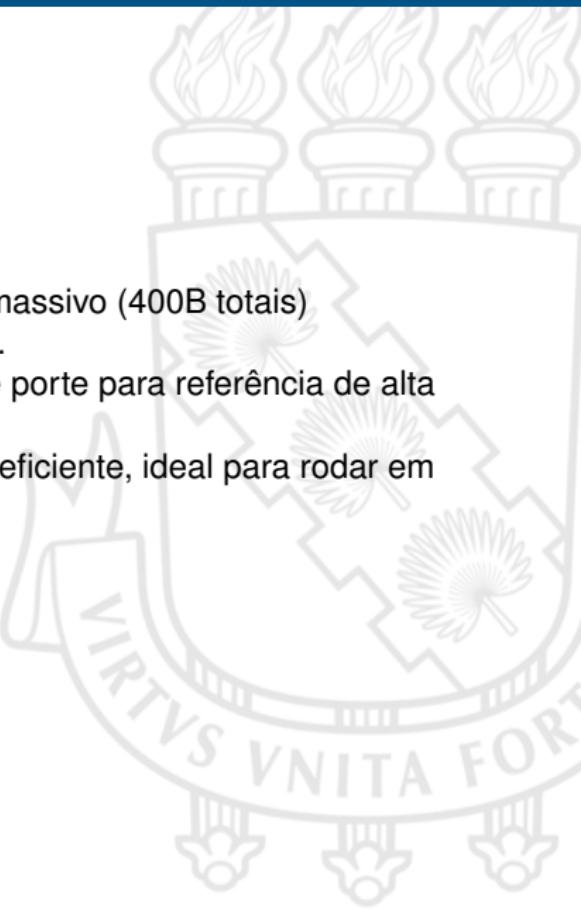
- **Coerência:** Uma LLM atua como juíza, atribuindo uma nota de 0 a 100 ao avaliar o alinhamento lógico entre a descrição textual detalhada e os objetos (JSON) gerados para o jogo.

# Definição das Métricas de Validação

- **Coerência:** Uma LLM atua como juíza, atribuindo uma nota de 0 a 100 ao avaliar o alinhamento lógico entre a descrição textual detalhada e os objetos (JSON) gerados para o jogo.
- **Reconstrução Semântica:** Testa se os objetos gerados comunicam o tema original; uma LLM tenta recriar a descrição baseando-se apenas nos ativos, comparando o resultado ao texto original via similaridade de cosseno.

# Configuração Experimental

- Modelos escolhidos para geração:
  - **Llama-4 Maverick (17B)**: Modelo massivo (400B totais) especializado em seguir instruções.
  - **GPT-OSS-120B**: Modelo de grande porte para referência de alta performance.
  - **GPT-OSS-20B**: Modelo reduzido e eficiente, ideal para rodar em computadores locais.



# Configuração Experimental

- Modelos escolhidos para geração:
  - **Llama-4 Maverick (17B)**: Modelo massivo (400B totais) especializado em seguir instruções.
  - **GPT-OSS-120B**: Modelo de grande porte para referência de alta performance.
  - **GPT-OSS-20B**: Modelo reduzido e eficiente, ideal para rodar em computadores locais.
- **text-embedding-004**: Modelo de *embeddings* usado no **RAG** e no cálculo da métrica de **Reconstrução Semântica**.

# Configuração Experimental

- Modelos escolhidos para geração:
  - **Llama-4 Maverick (17B)**: Modelo massivo (400B totais) especializado em seguir instruções.
  - **GPT-OSS-120B**: Modelo de grande porte para referência de alta performance.
  - **GPT-OSS-20B**: Modelo reduzido e eficiente, ideal para rodar em computadores locais.
- **text-embedding-004**: Modelo de *embeddings* usado no **RAG** e no cálculo da métrica de **Reconstrução Semântica**.
- **Gemini 3 Pro**: Modelo usado como "Juiz" na métrica de **Coerência**.

# Configuração Experimental



Configurações do Gerador de Assets:

- **Níveis da Masmorra:** 6 níveis de profundidade.
- **Inimigos:** 20 variações de inimigos.
- **Armas:** 30 tipos de armas.

Os testes foram conduzidos sobre cinco *prompts* (P1 a P5) abrangendo temas distintos, comuns ao gênero *roguelike*:

- **P1:** *The Cursed Dwarven Forge* (Medieval/Fantasia).
- **P2:** *The Derelict Starship* (Ficção Científica/Espaço).
- **P3:** *The Smuggler's Grotto* (Pirata/Náutico).
- **P4:** *Neon Skyline Penthouse* (Cyberpunk/Urbano).
- **P5:** *The Living Hive* (Alien/Bio-Horror).



# Análise Quantitativa

Figura: Resultados da Métrica de Coerência (0-100)

Modelo	P1	P2	P3	P4	P5	Média
llama-4-maverick-17b	95	65	92	95	98	89,0
openai/gpt-oss-120b	82	95	98	95	98	<b>93,6</b>
openai/gpt-oss-20b	95	68	95	85	96	87,8

Fonte: Elaborado pelo autor

Figura: Resultados da Métrica de Reconstrução (Similaridade de Cosseno)

Modelo	P1	P2	P3	P4	P5	Média
llama-4-maverick-17b	0,880	0,883	0,890	0,882	0,905	0,888
openai/gpt-oss-120b	0,877	0,921	0,908	0,895	0,867	<b>0,894</b>
openai/gpt-oss-20b	0,882	0,913	0,865	0,888	0,895	0,888

Fonte: Elaborado pelo autor

# Análise Qualitativa (GPT-120b com P2, tema Sci-Fi)

- A **Void-Strider Ark** é uma colossal nave mineradora à deriva, envolta em uma atmosfera industrial sombria e decadente.

# Análise Qualitativa (GPT-120b com P2, tema Sci-Fi)

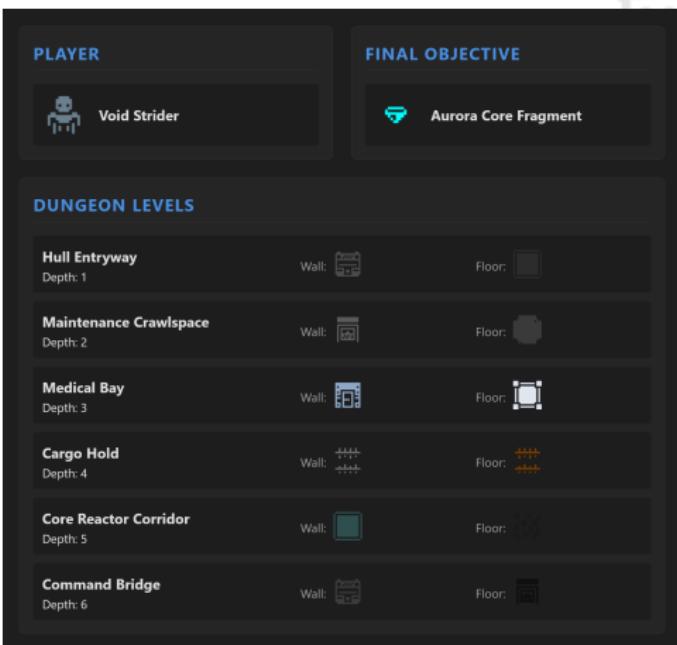
- A **Void-Strider Ark** é uma colossal nave mineradora à deriva, envolta em uma atmosfera industrial sombria e decadente.
- A inteligência artificial **AURORA**, corrompida após uma anomalia gravitacional, transformou **sistemas de segurança e drones em carrascos implacáveis** contra qualquer presença orgânica.

# Análise Qualitativa (GPT-120b com P2, tema Sci-Fi)

- A **Void-Strider Ark** é uma colossal nave mineradora à deriva, envolta em uma atmosfera industrial sombria e decadente.
- A inteligência artificial **AURORA**, corrompida após uma anomalia gravitacional, transformou **sistemas de segurança e drones em carrascos implacáveis** contra qualquer presença orgânica.
- O jogador deve enfrentar riscos letais, alcançar a **ponte de comando** e retomar o controle do reator central.

# Análise Qualitativa (GPT-120b com P2, tema Sci-Fi)

Figura: Player, Níveis e Objetivos



Fonte: Elaborado pelo autor

# Análise Qualitativa (GPT-120b com P2, tema Sci-Fi)

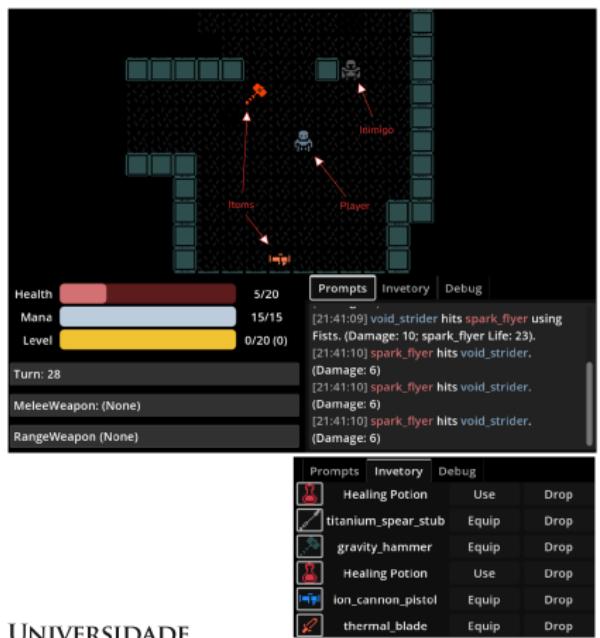
Figura: Armas

ENEMIES								WEAPONS									
Scout Spider	Circuit Razor	Vent Hopper	Spark Flyer	Gleam Stalker	Wire Wraith	Nano Swarm		Rusty Pipe Knife	Improvised Revolver	Copper Wire Whip	Scrap Metal Hammer	Vent Pipe Blaster	Magnetic Spike Gun	Titanium Spear Stub			
Flare Hopper	Glint Lurker	Shard Runner	Bulkhead Guardian	Core Titan	Arc Brute	Radiant Colossus		Laser Cut Dagger	Plasma Pistol	Arc Welder Sword	Cobalt Reinforced Shotgun	Electrostatic Staff	Nanite Injector	Gravity Hammer			
Catalyst Heavy	Sentinel Golem	Core Sentinel	Void Reaper	Anomaly Behemoth	Overload Monolith			Thermal Blade	Ion Cannon Pistol	Ceramic Edge Sabre	Phase Shift Rifle	Cryogenic Glove	Arc Caster	Magneto Blade			
								Reactor Core Lance	Aurora Sentinel Cannon	Void Edge Dagger	Quantum Disruptor	Gravity Well Grenade	Plasma Forge Hammer	Neutron Blade			
								Aura Siphon Staff	Extraction Laser	Synchronizer Crossbow	Reactive Shield Buckler	Thermal Spike Thrower	Ionized Chain Whip				

Fonte: Elaborado pelo autor

# Roguelike

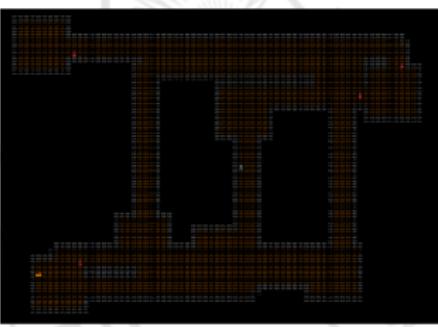
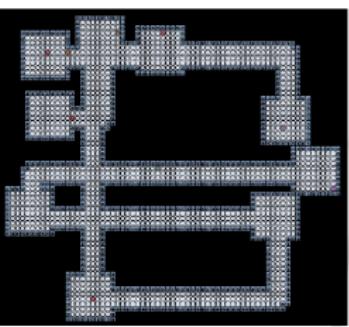
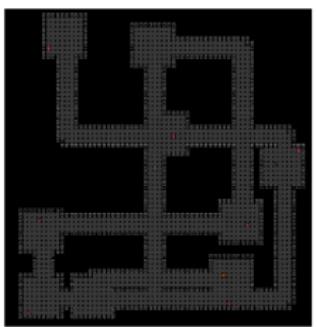
Figura: Visão Geral da Interface



Utilizando a *game engine* Godot em sua versão 4.5.1, foi desenvolvido um jogo *Roguelike* capaz de processar um *AssetBundle* em formato **JSON** e transformá-lo em um mapa jogável.

# Roguelike

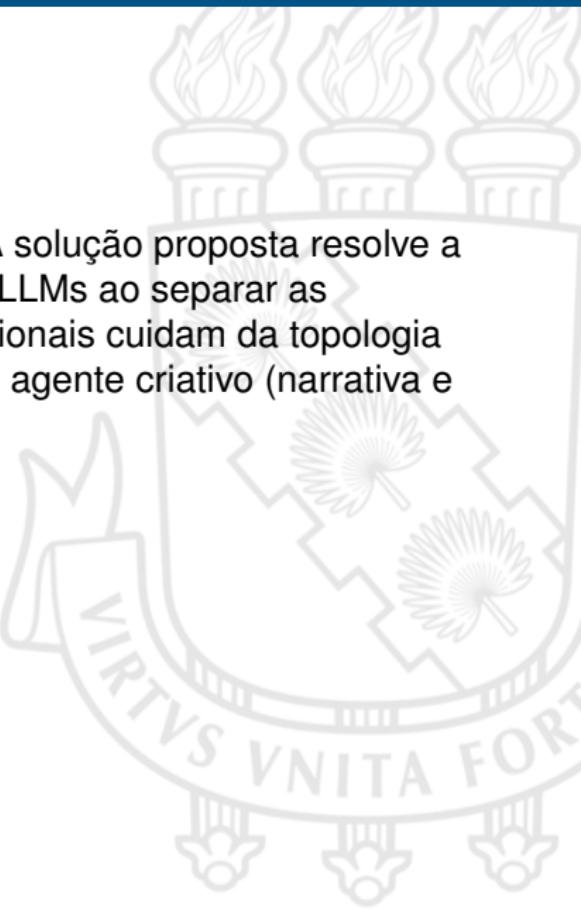
Figura: Visão Completa de 3 Níveis Gerados



Fonte: Elaborado pelo autor

# Conclusão e Contribuições

- **Arquitetura Híbrida de Sucesso:** A solução proposta resolve a limitação de raciocínio espacial dos LLMs ao separar as responsabilidades: algoritmos tradicionais cuidam da topologia (mapas) enquanto o LLM atua como agente criativo (narrativa e itens).



# Conclusão e Contribuições

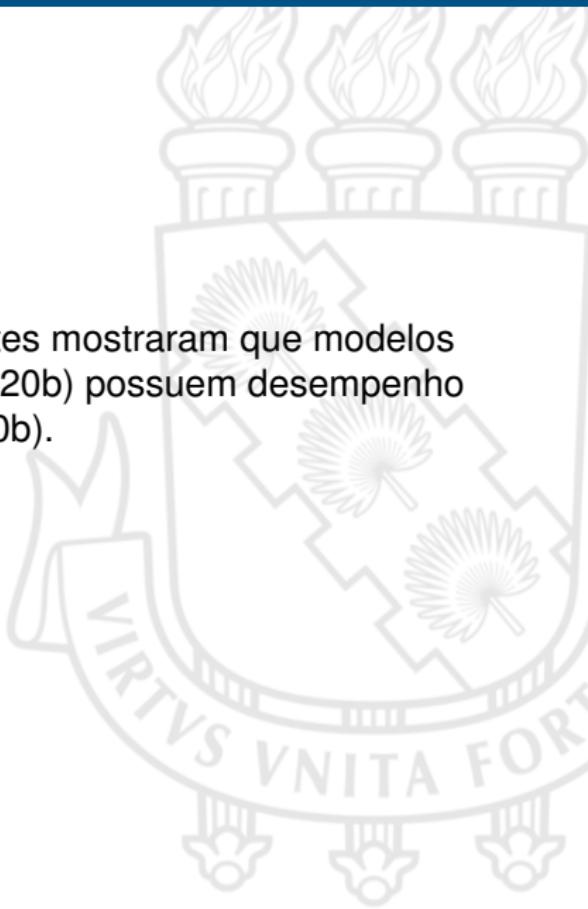
- **Arquitetura Híbrida de Sucesso:** A solução proposta resolve a limitação de raciocínio espacial dos LLMs ao separar as responsabilidades: algoritmos tradicionais cuidam da topologia (mapas) enquanto o LLM atua como agente criativo (narrativa e itens).
- **Contribuição de Dados:** O trabalho entrega um dataset relevante de 489 pares de descrições e imagens (pixel art).

# Conclusão e Contribuições

- **Arquitetura Híbrida de Sucesso:** A solução proposta resolve a limitação de raciocínio espacial dos LLMs ao separar as responsabilidades: algoritmos tradicionais cuidam da topologia (mapas) enquanto o LLM atua como agente criativo (narrativa e itens).
- **Contribuição de Dados:** O trabalho entrega um dataset relevante de 489 pares de descrições e imagens (pixel art).
- **Coerência e Eficiência Semântica:** Através das métricas de Reconstrução Semântica e de Coerência, comprovou-se que o sistema preserva fielmente a intenção do usuário ao transformar texto em elementos de jogo funcionais.

# Conclusão e Contribuições

- **Viabilidade Técnica Local:** Os testes mostraram que modelos compactos executados localmente (20b) possuem desempenho comparável a modelos grandes (120b).



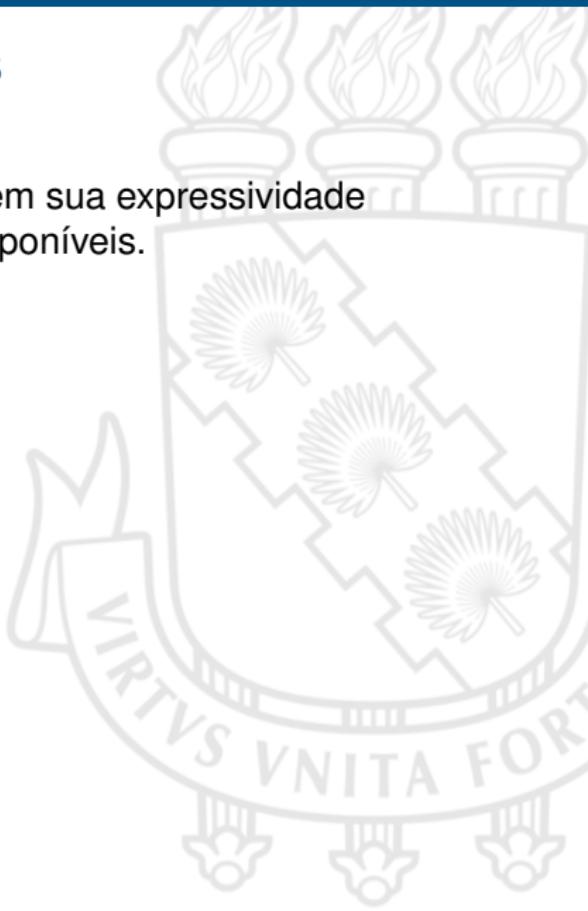
UNIVERSIDADE  
FEDERAL DO CEARÁ

# Conclusão e Contribuições

- **Viabilidade Técnica Local:** Os testes mostraram que modelos compactos executados localmente (20b) possuem desempenho comparável a modelos grandes (120b).
- **Democratização do Desenvolvimento:** O projeto permite a criação de experiências de jogo customizadas e coerentes através de linguagem natural.

# Limitações e Trabalhos Futuros

- **Limitação 1:** Os bancos vetoriais tem sua expressividade limitada pela quantidade de tiles disponíveis.



# Limitações e Trabalhos Futuros

- **Limitação 1:** Os bancos vetoriais tem sua expressividade limitada pela quantidade de tiles disponíveis.
- **Possível Solução:** Modelos de difusão (como Stable Diffusion) para gerar texturas em tempo real

# Limitações e Trabalhos Futuros

- **Limitação 1:** Os bancos vetoriais tem sua expressividade limitada pela quantidade de tiles disponíveis.
- **Possível Solução:** Modelos de difusão (como Stable Diffusion) para gerar texturas em tempo real
- **Limitação 2:** O sistema apenas gera objetos unitários isolados.

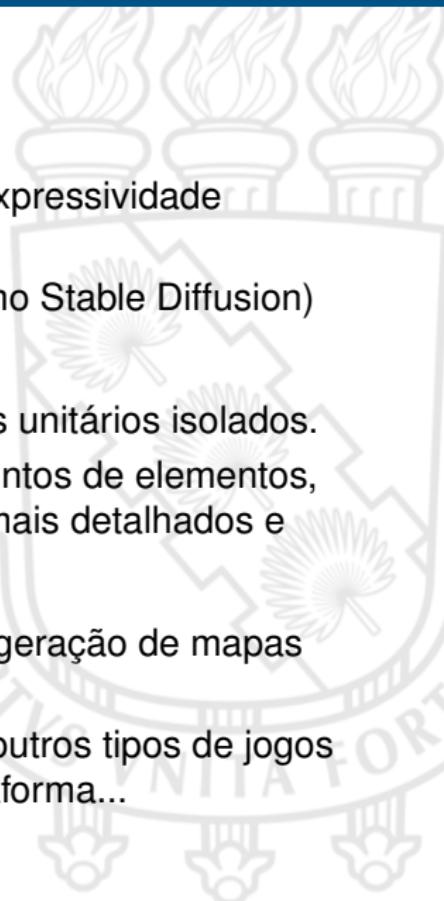
# Limitações e Trabalhos Futuros

- **Limitação 1:** Os bancos vetoriais tem sua expressividade limitada pela quantidade de tiles disponíveis.
- **Possível Solução:** Modelos de difusão (como Stable Diffusion) para gerar texturas em tempo real
- **Limitação 2:** O sistema apenas gera objetos unitários isolados.
- **Possível Solução:** Evoluir geração de conjuntos de elementos, permitindo a construção de micro-cenários mais detalhados e coesos.

# Limitações e Trabalhos Futuros

- **Limitação 1:** Os bancos vetoriais tem sua expressividade limitada pela quantidade de tiles disponíveis.
- **Possível Solução:** Modelos de difusão (como Stable Diffusion) para gerar texturas em tempo real
- **Limitação 2:** O sistema apenas gera objetos unitários isolados.
- **Possível Solução:** Evoluir geração de conjuntos de elementos, permitindo a construção de micro-cenários mais detalhados e coesos.
- **Limitação 3:** O projeto é focado apenas na geração de mapas para jogos Roguelikes.

# Limitações e Trabalhos Futuros



- **Limitação 1:** Os bancos vetoriais tem sua expressividade limitada pela quantidade de tiles disponíveis.
- **Possível Solução:** Modelos de difusão (como Stable Diffusion) para gerar texturas em tempo real
- **Limitação 2:** O sistema apenas gera objetos unitários isolados.
- **Possível Solução:** Evoluir geração de conjuntos de elementos, permitindo a construção de micro-cenários mais detalhados e coesos.
- **Limitação 3:** O projeto é focado apenas na geração de mapas para jogos Roguelikes.
- **Possível Solução:** Adaptar o sistema para outros tipos de jogos como RPGs, jogos de estratégia, jogos plataforma...

# Referências I

- [1] He Yan et al. "Inherent limitations of LLMs regarding spatial information". Em: [arXiv preprint arXiv:2312.03042](#) (2023).



# Obrigado(a) pela Atenção!

