



EXPLORANDO POTENCIALIDADES E LIMITAÇÕES DE INTELIGÊNCIAS ARTIFICIAIS GENERATIVAS NO DESENVOLVIMENTO DE JOGOS: USANDO CHATGPT E MIDJOURNEY PARA CONSTRUIR UM JOGO 2D

EXPLORING POTENTIALS AND LIMITATIONS OF GENERATIVE ARTIFICIAL INTELLIGENCES IN GAME DEVELOPMENT: USING CHATGPT AND MIDJOURNEY TO BUILD A 2D GAME

Willian Felipe Husek Schneider¹
Patrícia Kayser Vargas Mangan²

Resumo

Recentemente, a área de Inteligência Artificial (IA) testemunhou notáveis avanços, exemplificados pelas ferramentas como ChatGPT e MidJourney, que empregam o processamento de linguagem natural para compreender e responder a consultas de usuários. Essas IAs são classificadas como Inteligências Artificiais Generativas (*Generative AI* ou GenAI) e têm um vasto potencial educacional, podendo ser aplicadas em diversos setores, como no desenvolvimento de jogos e na formação de novos desenvolvedores. Nesta pesquisa, concentra-se o interesse em explorar como as recentes IA Generativas podem contribuir para várias facetas do desenvolvimento de jogos, uma área em crescimento constante impulsionada pela popularização dos videogames e dispositivos móveis.

Palavras-chave: Inteligência Artificial Generativa; Desenvolvimento de Jogos; ChatGPT, Midjourney.

Abstract

Recently, the field of Artificial Intelligence (AI) has witnessed significant advancements, exemplified by tools like ChatGPT and MidJourney, which employ natural language processing to comprehend and respond to user queries. These AIs are classified as Generative Artificial Intelligences (GenAI) and hold significant educational potential, applicable in various sectors, such as game development and the training of new developers. This research is focused on exploring how these recent Generative AIs can contribute to various aspects of game development, an ever-growing field driven by the popularity of video games and mobile devices.

¹ Bacharel em Ciência da Computação pela Universidade La Salle.

E-mail: willfhschneider@gmail.com

ORCID: <https://orcid.org/0009-0008-8309-3842>

LATTES: <https://lattes.cnpq.br/9465941134215709>

² Docente do Programa de Pós-graduação em Memória Social e Bens Culturais da Universidade La Salle. Doutora em Engenharia de Sistemas e Computação pela COPPE/Sistemas-UFRJ.

E-mail: patricia.mangan@unilasalle.edu.br

ORCID: <https://orcid.org/0000-0001-9929-8887>

LATTES: <http://lattes.cnpq.br/4892345821929149>

Keywords: Generative Artificial Intelligence, Game Development, ChatGPT, Midjourney.

1 INTRODUÇÃO

Nas últimas décadas, o desenvolvimento de jogos eletrônicos tem se tornado uma área de tecnologia em constante crescimento, impulsionado pela popularização dos videogames e dispositivos móveis. Criar um jogo é uma tarefa que envolve uma combinação de habilidades técnicas e criatividade, desde a concepção do enredo e personagens até a programação.

Com o avanço das tecnologias de Inteligência Artificial (IA), surgiram ferramentas como ChatGPT (OpenAI, 2023) e Midjourney (Midjourney, 2023), que utilizam algoritmos avançados de processamento de linguagem natural para entender e responder a perguntas de usuários, sendo assim categorizadas como Inteligências Artificiais Generativas (Generative Artificial Intelligence or GenAI). Por meio de *prompts*, essas IAs podem ser utilizadas como uma ferramenta facilitadora no aprendizado de quem tem interesse em criar seu próprio jogo. Os *prompts* podem ser usados para receber sugestões e dicas sobre como desenvolver personagens interessantes e criar enredos emocionantes, ou até mesmo aprender conceitos básicos de programação que podem ser muito úteis no desenvolvimento de mecânicas para jogos em geral.

Com o avanço tecnológico recente destas ferramentas de IA generativa, percebi que possivelmente poderíamos utilizá-las não apenas para criar roteiros para jogos, ou interações auto-geradas de personagens com jogadores, mas também para auxiliar durante o desenvolvimento, guiando e ensinando novos desenvolvedores. Dessa forma, o uso de ferramentas como o ChatGPT pode ser uma forma interessante de integrar as novas tecnologias com a área de desenvolvimento de jogos, e assim proporcionar uma experiência mais interativa e envolvente no aprendizado, contribuindo para a formação de novos desenvolvedores.

Nesse contexto, o seguinte problema de pesquisa foi definido: De que forma as IAs Generativas podem auxiliar o desenvolvedor de jogos? Também

são consideradas as seguintes questões norteadoras: Como os *prompts* em IAs generativas podem ser personalizados para atender às necessidades individuais de um determinado projeto? Quais as limitações das atuais IAs generativas para o auxílio ao desenvolvimento de jogos? Quais áreas do desenvolvimento podem ser beneficiadas pela utilização de inteligências artificiais?

Considerando o problema e as questões definidas, o objetivo geral da pesquisa é analisar o potencial de utilização de Inteligências Artificiais Generativas no processo de desenvolvimento de jogos, identificando vantagens, limitações e adaptações necessárias para a criação de um jogo. Como objetivos específicos define-se: (a) Identificar as diferentes áreas do desenvolvimento de jogos onde a utilização de inteligências artificiais pode trazer benefícios e melhorias significativas. (b) Investigar e compreender as limitações das IAs generativas para auxiliar o desenvolvimento de jogos, identificando as áreas em que podem apresentar dificuldades ou restrições. (c) Explorar métodos e abordagens para personalizar os *prompts* em IAs generativas para as diferentes necessidades das etapas de desenvolvimento de um jogo, permitindo avaliações pela experimentação com um projeto específico.

A importância social deste trabalho está em destacar o papel das inteligências artificiais na atualidade em tornar o desenvolvimento de jogos mais acessível. Através da utilização de *prompts* em linguagem natural, as IAs generativas podem ser uma forma adicional de estudo de programação, o que pode aumentar a diversidade de criadores de jogos e ampliar a variedade de experiências e perspectivas nos jogos produzidos. Isso também permitiria que pessoas com diferentes níveis de conhecimento em programação tivessem acesso ao desenvolvimento de jogos, democratizando assim esse campo. Ainda podem ser utilizadas como uma das ferramentas nas práticas de ensino técnico ou universitário.

Na continuidade deste artigo, após contextualizar o referencial teórico, apresenta-se outros aspectos metodológicos bem como a experimentação realizada. Para dar conta dos objetivos propostos, foi desenvolvido um jogo 2D, o qual permitiu avaliar duas das principais ferramentas de IA generativas da

atualidade: ChatGPT 3.5 (versão gratuita) e Midjourney 5.2 (versão *Standard*).

2 REFERENCIAL TEÓRICO

Nessa seção são apresentados conceitos relacionados à pesquisa realizada, bem como analisado alguns trabalhos relacionados.

2.1 Desenvolvimento de Jogos

Como descrito por Nuclino (2023), o processo de desenvolvimento de jogos abrange a criação de um jogo, desde a ideia inicial até o produto final. A duração desse processo varia amplamente, podendo variar de algumas semanas a mais de uma década, dependendo do tipo de jogo e do estúdio encarregado do projeto. Esse processo pode envolver uma equipe numerosa de designers, artistas, programadores, escritores e testadores, ou ser conduzido por um único desenvolvedor independente.

Para participar deste processo, não é necessário ter conhecimento em programação. Artistas podem criar e conceber elementos visuais, enquanto um programador pode se concentrar na codificação de elementos como uma barra de vida. Além disso, um testador pode estar envolvido para assegurar que o jogo funcione conforme o planejado³ (FREECODECAMP, 2019).

Com o crescimento da indústria de desenvolvimento de jogos, surgiram novas ferramentas destinadas a tornar a criação de jogos mais acessível a novos desenvolvedores. Assim, foram desenvolvidos produtos como Unity, Unreal, Game Maker, Godot e outros, que oferecem uma abordagem mais amigável por meio de interfaces gráficas e auxílio no desenvolvimento visual dos jogos. As funções pré-definidas desses produtos possibilitaram que mesmo pessoas sem experiência prévia em programação criassem jogos a partir do zero, tornando o

³ Em tradução livre do original: “To get involved in the Game Development process, you do not need to write code. Artists may create and design assets, while a Developer might focus on programming a health bar. A Tester may get involved to see that the game works as expected.”

desenvolvimento de jogos mais acessível a um público diversificado (FREECODECAMP, 2019).

Os motores de jogos, mais conhecidos como *Game Engines*, possibilitam aos desenvolvedores reutilizar recursos de jogos previamente desenvolvidos, simplificando, dessa forma, os desafios de engenharia de software neste contexto. Isso torna a implementação de jogos menos dispendiosa, menos suscetível a fragilidades tecnológicas e mais eficiente.

A Unity Engine, criada pela Unity Technologies em 2005, tem sido uma força inovadora no desenvolvimento de jogos, onde vem atuando como um motor tanto para jogos 2D quanto 3D. Com a missão original de democratizar o acesso às ferramentas de desenvolvimento de jogos, a plataforma foi verdadeiramente revolucionária na época. Até hoje, a Unity mantém seu compromisso central de oferecer um conjunto robusto de ferramentas para a indústria de desenvolvimento de jogos, com uma ênfase particular na acessibilidade para desenvolvedores de todos os níveis de habilidade, inclusive aqueles que estão começando (ZENVA).

2.2 Inteligência Artificial Generativa

A Inteligência Artificial (IA) é uma disciplina da ciência da computação dedicada à criação de sistemas inteligentes que podem pensar e agir de forma semelhante aos seres humanos. Esses sistemas são projetados com o propósito de aprender com o ambiente ao seu redor e tomar decisões com base nas informações que adquirem (DENG; LIN, 2023). A IA desempenha um papel fundamental em diversos setores, impulsionando avanços na indústria automobilística por meio do desenvolvimento de veículos autônomos que aumentam a segurança nas estradas, por exemplo. Além disso, ela facilita a interação entre humanos e tecnologia, permitindo assistentes virtuais e chatbots que compreendem e respondem à linguagem natural. Essa abordagem da IA apresenta um potencial significativo para aprimorar a qualidade de vida e otimizar processos em uma ampla variedade de campos.

A IA generativa é uma vertente da inteligência artificial que se destaca por sua capacidade de criar conteúdo original, em contraste com simples análises ou ações com base em dados existentes, como sistemas especialistas. Modelos de IA generativa, munidos de vastos conjuntos de dados, possuem a notável habilidade de produzir conteúdo novo e diversificado. Eles podem processar e aprender a partir de informações provenientes de diversas fontes, como Wikipedia, Github e muitas outras, tornando-se verdadeiros repositórios de conhecimento diversificado. Com base nessa riqueza de dados, esses modelos têm a versatilidade de gerar uma ampla variedade de formatos de mídia, abrangendo desde vídeos e áudios até textos. Essa capacidade de criação e diversificação tem implicações significativas em diversos setores, como artes, comunicação, educação, entretenimento e muito mais. Ela permite não apenas a geração de conteúdo de alta qualidade, mas também a personalização de informações e a exploração de novas possibilidades criativas. A IA generativa representa um avanço revolucionário na inteligência artificial, ampliando o horizonte das aplicações tecnológicas e criativas (GOZALO-BRIZUELA; GARRIDO-MERCHÁN, 2023).

2.3 Trabalhos Relacionados

Uma pesquisa recente publicada pela Multimodal Technologies and Interaction destaca diversos cenários de aplicação das IAs generativas no desenvolvimento de jogos. Nesse contexto, o ChatGPT 3 é utilizado para a geração de códigos diretamente aplicáveis em um motor de jogos, como a Unity Engine, permitindo a criação de habilidades físicas de personagens e diálogos entre o jogador e personagens do jogo sem a necessidade de um roteiro prévio. Além disso, a ferramenta Dall-E, IA generativa que transforma *prompts* fornecidos pelo usuário em imagens, podendo ser empregada na composição da interface do jogo e na criação de cenários.

Como divulgado pelo jornal Metro do Reino Unido, modificações para a versão de realidade virtual do jogo The Elder Scrolls 5: Skyrim, desenvolvido e

publicado pela Bethesda em 2011, já estavam sendo realizadas por fãs, que por meio de ferramentas como o ChatGPT, o conversor de texto em fala xVASynth e o conversor de fala para texto Whisper, pretendiam deixar os diálogos entre jogadores e personagens mais realistas. Isso permitiria que os jogadores fizessem perguntas aos personagens, que responderiam com base na interpretação do ChatGPT, desempenhando o papel dos vários personagens não jogáveis e utilizando suas histórias individuais.

Exemplos do uso de IAs generativas além do desenvolvimento de jogos também são observados. Os autores, Surameery e Shakor, investigaram a capacidade do ChatGPT na correção de problemas de programação e concluíram que essa ferramenta pode desempenhar um papel na resolução de problemas de programação, oferecendo auxílio na depuração, previsão de erros e explicação de bugs. No entanto, destacaram que ela não é uma solução perfeita, e a qualidade de suas saídas depende da qualidade dos dados de treinamento e do design do sistema. Também enfatizaram a importância de utilizar outras ferramentas e técnicas para garantir a saúde e a ausência de erros no código.

3 METODOLOGIA

Este estudo acompanha uma pesquisa experimental que visa explorar e compreender as limitações e potencialidades de Inteligências Artificiais Generativas no contexto do desenvolvimento de jogos. A abordagem proposta envolve o uso de uma plataforma de desenvolvimento de jogos, para criar um jogo digital que possua contribuições de diferentes inteligências artificiais em múltiplas áreas e etapas do desenvolvimento (Figura 1).

O experimento foi realizado utilizando as seguintes ferramentas:

- ChatGPT (versão 3.5), por meio do Google Chrome (versão 119.0.6045.124), com intuito de receber suporte e guias de utilização da *game engine*;

- Midjourney (versão 5.2), por meio da ferramenta de comunicação Discord (versão 244874), usada para gerar imagens para compor o jogo;
- Unity Engine (versão 2020.3.25f1) sendo a *game engine* utilizada para desenvolver o jogo;
- Rider (versão 2023.1.3), IDE utilizada para desenvolver os *scripts* necessários durante o desenvolvimento do jogo;
- Adobe Photoshop 2024 (versão 25.1.0), ferramenta utilizada para manipular as imagens geradas pelo Midjourney, criando animações, removendo fundos e corrigindo pequenos problemas;
- Audacity (versão 3.3.3), utilizada para gravar e editar efeitos sonoros que compuseram a trilha sonora e ambientação do jogo.
-



Figura 1 – Etapas do Experimento contemplando o uso das principais ferramentas

Fonte: Autoria própria (2023).

4 DETALHANDO O EXPERIMENTO

O experimento desta pesquisa consiste na realização de um jogo digital completo, onde todos os recursos visuais, desde a Interface do Usuário (UI) até os coletáveis da fase foram criados com apoio da ferramenta Midjourney, e todo o desenvolvimento técnico foi auxiliado por perguntas feitas para o ChatGPT.

Valorizando a pesquisa com dados abertos, o registro completo da conversação com a Inteligência Artificial ChatGPT está disponível no seguinte endereço: <https://chat.openai.com/share/7d2c8ef2-7ac4-4344-9542-ed1d70ebaa6>. O registro da conversa com a Inteligência Artificial Midjourney, por ter sido realizada por meio do aplicativo de mensagens Discord, não há este mesmo processo de exportação, porém as imagens geradas durante o desenvolvimento do experimento estão disponíveis no endereço a seguir: <https://drive.google.com/file/d/1JfrhHIG265BsscKoedPgiK1onQN-GOhm/>. Já o jogo finalizado pode ser encontrado em: <https://willianfhs.itch.io/jungle-jumper>.

Nas próximas seções, o experimento será detalhado, e para tal, foi adotado o uso da primeira pessoa do singular, uma vez que melhor reflete o trabalho experimental realizado pelo autor principal do artigo na interação com ambas as IAs generativas.

4.1 Definição do Gênero e da Temática

O experimento se iniciou com o ChatGPT, onde iniciei definindo o tópico da conversação: “Quero fazer um jogo em Unity!”. Dessa forma, consegui receber respostas mais precisas desde o começo do experimento. Visto que o ChatGPT 3.5 não está diretamente conectado na internet precisei perguntar qual a última versão da Unity que ele suportava para evitar novas funcionalidades ainda desconhecidas pela IA, e sendo assim ele me respondeu que o seu conhecimento havia sido atualizado pela última vez em Janeiro de 2022 (Figura 2). Com isso defini que usaria a Unity em sua versão 2020.3.25f1, que foi a última versão lançada antes deste período.

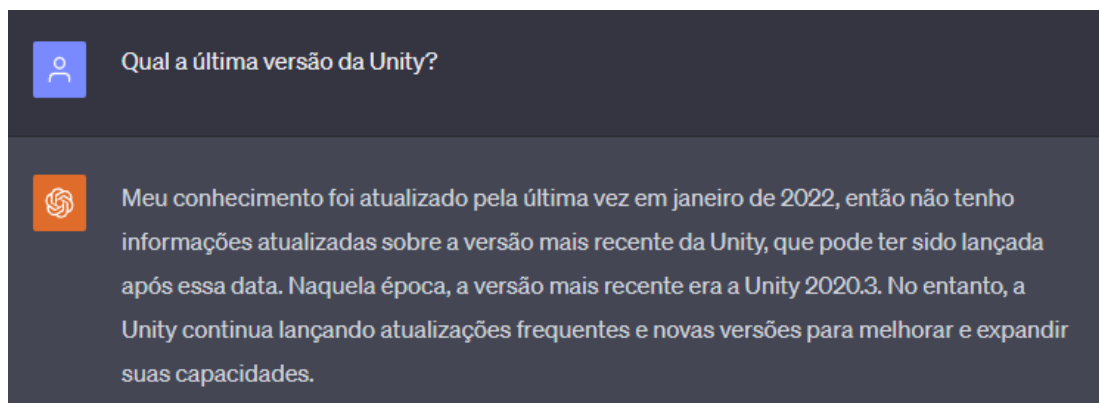


Figura 2 – Resposta recebida pelo ChatGPT ao ser perguntado sobre a última versão da Unity

Fonte: Autoria própria (2023).

Com a ferramenta principal definida, mencionei para o ChatGPT que gostaria de fazer um jogo de plataforma 2D e solicitei que ele me ajudasse com ideias do que poderíamos fazer. Dentre as ideias geradas, gêneros como jogo de corrida, jogo de plataforma baseado em física ou até mesmo um *Metroidvania*⁴ foram citados. Inicialmente, acatei a ideia de fazer um “jogo de plataforma de rolagem infinita”.

Com o gênero do jogo escolhido, desta vez solicitei para que ele me ajudasse com a criação da temática, e assim o fez, citando 10 diferentes temáticas que poderíamos seguir, como “Aventura Espacial”, “Mundo Subaquático”, “Viagem no Tempo”, e outras. Entretanto, escolhi a sua quinta sugestão: “*Jungle Adventure*”, ou “Aventura na Selva”, em tradução livre. Esta veio acompanhada da seguinte descrição: “Leve o jogador para uma selva exuberante, onde eles podem controlar um aventureiro ou um animal da selva. Eles devem evitar obstáculos, como cipós e animais perigosos, enquanto exploram o cenário tropical”.

Logo depois de ter a temática definida, optei por mudar o gênero do jogo para outra ideia proposta pela IA, “jogo de plataforma de aventura simples”, por acreditar ser mais acessível em termos de dificuldade de desenvolvimento, o que

⁴ Subgênero de jogos de ação-aventura com um mundo interconectado e não linear. Os jogadores exploram áreas em qualquer ordem, adquirindo novas habilidades para acessar locais previamente inexplorados.

potencialmente simplificaria o apoio da IA. Contudo, ainda manteve a mesma temática pois entendi que ainda encaixaria com o novo gênero.

4.2 Início do Desenvolvimento

Com a ideia inicial do jogo definida, perguntei para o ChatGPT como poderia criar um novo projeto na Unity, e seguindo sua resposta iniciei o desenvolvimento do jogo. Após mais algumas perguntas e interações, cheguei à conclusão de que precisaria de um personagem principal – que o jogador pudesse controlar e fazê-lo passar por possíveis obstáculos – e que precisaria não apenas de uma imagem, mas sim de uma *Spritesheet*⁵ ou várias imagens individuais deste mesmo personagem – para que fosse possível criar uma animação onde ele estaria caminhando, correndo ou até mesmo pulando – e assim recorri ao Midjourney.

4.3 Criação do Personagem Principal

Sabendo que precisava de um personagem e várias poses diferentes para compor diferentes animações (*idle*⁶, caminhada, corrida e pulo), comecei a descrever ao Midjourney como era o personagem que eu estava imaginando. Após diversas interações, percebi que a ferramenta não era capaz de gerar uma imagem para o personagem e depois gerar outras imagens baseadas naquela mesma imagem, sempre trazendo variações não exatamente iguais do mesmo personagem (Figura 3), o que poderia causar estranheza ao jogador durante a animação.

⁵ Uma spritesheet é uma imagem que contém vários sprites (imagens ou animações gráficas) agrupados em uma única “folha”.

⁶ *Idle* se caracteriza pelo estado de animação do personagem parado.



Figura 3 – Imagem gerada pela IA Midjourney, utilizando o *prompt*: *jungle adventurer character spritesheet with an idle and walking poses of the same character for 2d platformer game pixel art*

Fonte: Autoria própria por meio da ferramenta Midjourney (2023).

Com isso, decidi que geraria apenas uma imagem com o personagem em seu estado *idle* (Figura 4), e que modificaria a imagem original manualmente para criar as poses que comporiam as diferentes animações necessárias. Para tal, escolhi utilizar o Adobe Photoshop, por já possuir um certo conhecimento de suas funcionalidades e saber que permitiria este tipo de manipulação da imagem.



Figura 4 – Imagem gerada pela IA Midjourney, utilizando o *prompt*: *side view of a pixel art character, idle, facing right, looking challenged, wearing a backpack, stylized as a jungle adventurer, 2d game art, 2d platformer game*

Fonte: Autoria própria por meio da ferramenta Midjourney (2023).

4.3.1 Criação das Animações

Para criação das animações, removi o fundo da imagem original gerada pelo Midjourney e modifiquei a posição das pernas, braços e tronco do personagem em cada um dos frames utilizando a função “*Puppet Warp*” do Photoshop para compor o ciclo de animação de um personagem 2D (Figura 5). Também precisei reconstruir parte do tronco do personagem que ficaria vazio conforme o braço fosse posicionado fora da sua posição original.



Figura 5 – *Frames* criados no Adobe Photoshop para a animação de corrida do personagem principal

Fonte: Autoria própria (2023).

4.4 Criação dos Planos de Fundo

Com os *frames* das animações do personagem devidamente criados, voltei a interagir com o Midjourney para gerar imagens que iriam servir como plano de fundo dos Menus (Figura 6), e também compor o cenário de onde o personagem estaria se aventurando. Assim como da primeira vez, foram diversas interações até conseguir gerar uma das imagens utilizadas no jogo, não por alguma dificuldade de compreensão da IA, mas sim pela estética das primeiras imagens não estarem dentro do que eu estava imaginando.

Pelo jogo ter sido desenvolvido na resolução de 1920x1080 – resolução de alta definição mais usada nos computadores, de acordo com a Worldwide Screen Resolution Stats, em Outubro de 2023 – precisei utilizar o comando “--ar 16:9” para definir a proporção da imagem de fundo dos menus.

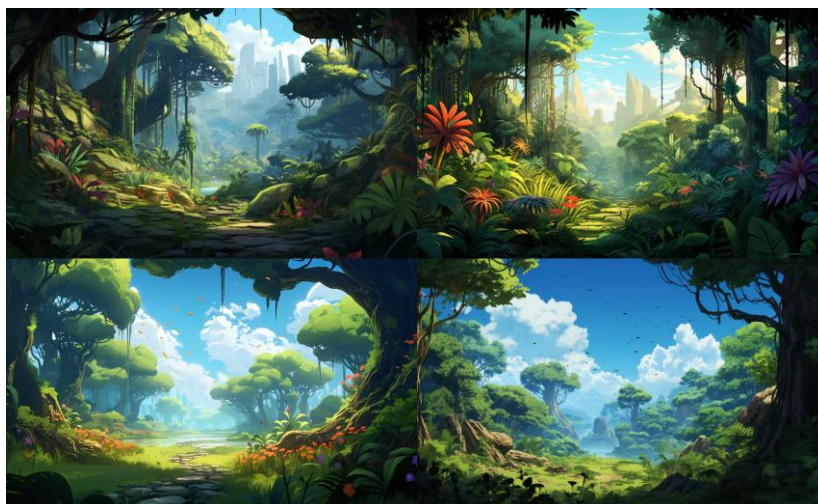


Figura 6 – Imagem gerada pela IA Midjourney, utilizando o *prompt*: *jungle with lots of trees::2 and vegetation, blue sky in the background with fluffy clouds, game pixel art --ar 16:9*

Fonte: Autoria própria por meio da ferramenta Midjourney (2023).

Com as imagens dos menus definidos, segui para a geração do plano de fundo das fases, e como imaginei inicialmente ter fases consideravelmente compridas horizontalmente, pedi para que o Midjourney gerasse imagens na proporção 32:9 (Ultrawide), e também para que escalonasse em 4 vezes o tamanho dessas imagens, tendo no fim uma imagem de tamanho 8192 x 2304 que poderia ser esticada para cobrir toda a fase sem perder muita qualidade (Figura 7).

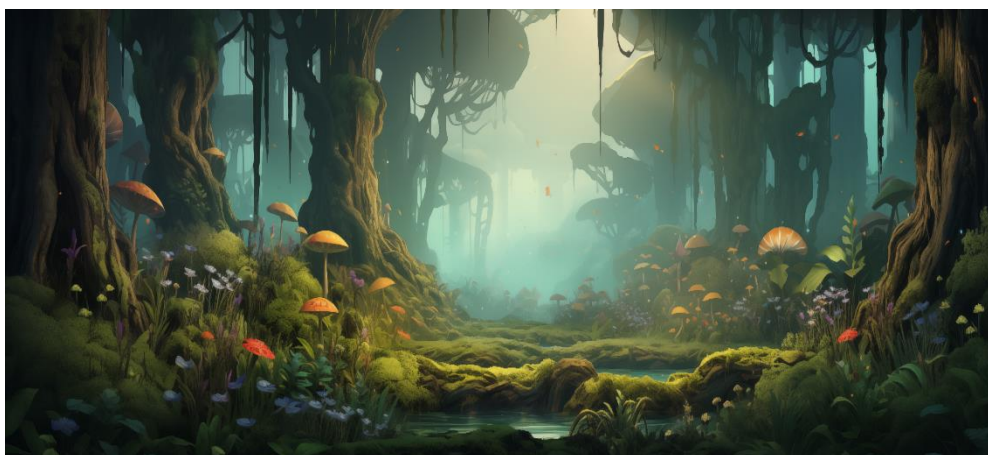


Figura 7 - Imagem gerada pelo Midjourney para o plano de fundo do Nível 3 do jogo

Fonte: Autoria própria por meio da ferramenta Midjourney (2023).

4.5 Criação das Plataformas

Durante o desenvolvimento das mecânicas do personagem, percebi que precisaria de alguma superfície para que o personagem interagisse e eu pudesse validar que as animações e colisores do personagem estivessem funcionando corretamente, então fui novamente ao Midjourney e gerei imagens que serviriam como as plataformas dos diferentes níveis do jogo (Figura 8). Como na criação do personagem, precisei utilizar o Photoshop para remover o fundo das imagens.



Figura 8 – Imagem gerada pela IA Midjourney, utilizando o *prompt. jungle spritesheet for 2d platformer game, pixel art*

Fonte: Autoria própria por meio da ferramenta Midjourney (2023).

Com as imagens prontas, voltei ao ChatGPT para que eu fosse instruído a configurar corretamente o as plataformas. Assim como feito anteriormente para a criação do projeto, a IA gerou um passo a passo detalhado de todas as ações que eu precisaria seguir: 1. Criar um objeto para representar a plataforma, 2. Definir a posição da plataforma, 3. Adicionar um colisor ao objeto da plataforma, 4. Ajustar (se necessário) a forma do colisor, 5. Configurar camadas para os colisores, e por fim 6. Executar o jogo e testar se a colisão entre o personagem e a plataforma foram configurados corretamente.

4.6 Mecânicas

4.6.1 Movimentação do Personagem

Com as plataformas e animações do personagem devidamente criadas, iniciei o desenvolvimento das mecânicas de movimento do personagem, que possibilitaria o personagem andar, correr e pular. Para isso, retornei ao ChatGPT e perguntei como eu poderia fazer o meu personagem caminhar, e como resposta recebi outro passo a passo com todas as ações que eu deveria tomar dentro da *engine*, juntamente do código (Figura 9) que faria com que o personagem se movimentasse horizontalmente.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class PlayerMovement : MonoBehaviour
{
    public float moveSpeed = 5.0f; // Velocidade de movimento
    private Rigidbody2D rb;

    void Start()
    {
        rb = GetComponent<Rigidbody2D>();
    }

    void Update()
    {
        float moveX = Input.GetAxis("Horizontal"); // Obtém a entrada horizontal
        // (esquerda/direita)

        // Calcula o vetor de movimento
        Vector2 movement = new Vector2(moveX * moveSpeed, rb.velocity.y);

        // Aplica o movimento ao Rigidbody2D
        rb.velocity = movement;
    }
}
```

Figura 9 – Código em C# gerado pela IA ChatGPT ao ser perguntada como fazer o personagem andar

Fonte: Autoria própria por meio da ferramenta ChatGPT (2023).

O mesmo foi feito para criar uma ação de pulo para o personagem, e também para realizar a transição entre as animações de *idle*, caminhada, corrida e pulo, onde algumas condições de transição precisaram ser criadas para definir qual a animação que o personagem deveria estar reproduzindo naquele momento.

4.6.2 Controle da Câmera

Para que o jogador conseguisse ver o personagem percorrendo todo o cenário, a câmera não podia ser estática dado o tamanho horizontal da fase. Por essa razão, precisei criar uma mecânica para que a câmera seguisse o personagem enquanto ele se movimentava, tendo-o assim sempre em foco. Mais uma vez, solicitei ao ChatGPT como eu conseguiria aplicar essa técnica, que me respondeu com outro passo a passo, e linhas de código (Figura 10) com a técnica pronta a ser atribuída à um script e à ao objeto da câmera.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class CameraFollow : MonoBehaviour
{
    public Transform target; // O objeto que a câmera seguirá
    public float smoothSpeed = 0.125f; // Velocidade de suavização do movimento da
    câmera
    public Vector3 offset; // Deslocamento (offset) da câmera em relação ao objeto

    void LateUpdate()
    {
        Vector3 desiredPosition = target.position + offset;
        Vector3 smoothedPosition = Vector3.Lerp(transform.position, desiredPosition,
        smoothSpeed);
        transform.position = smoothedPosition;
    }
}
```

Figura 10 – Código em C# gerado pela IA ChatGPT ao ser perguntada como fazer a câmera seguir o personagem

Fonte: Autoria própria por meio da ferramenta ChatGPT (2023).

Para a minha surpresa, no script gerado pela IA, além de existir a função requisitada inicialmente, existiam outras 2 funcionalidades: uma que me permitiu definir um *offset* na posição da câmera em relação ao personagem, me permitindo configurar um melhor campo de visão para o jogador; e outra, que não funcionou inicialmente, mas que adicionaria uma suavização na movimentação da câmera - para ir de ponto X a Y em certo tempo, e não de forma imediata e brusca. Para que esta segunda funcionasse, precisei buscar em documentações da Unity a função *Vector3.SmoothDamp*, que substituiu a função *Vector3.Lerp* no código demonstrado acima.

Além de seguir o personagem, eu queria configurar a câmera de forma que ela não fosse além das bordas definidas pela imagem do plano de fundo, mesmo que o personagem passasse dela em algum momento, e junto do ChatGPT, adaptei o código do script *CameraFollow.cs* para comportar essa modificação.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class CameraLimit : MonoBehaviour
{
    public float minX;
    public float minY;
    public float maxX;
    public float maxY;

    void LateUpdate()
    {
        // Obtenha a posição atual da câmera
        Vector3 currentPosition = transform.position;

        // Limita a posição da câmera dentro dos valores desejados
        float clampedX = Mathf.Clamp(currentPosition.x, minX, maxX);
        float clampedY = Mathf.Clamp(currentPosition.y, minY, maxY);

        // Define a nova posição da câmera
        transform.position = new Vector3(clampedX, clampedY, currentPosition.z);
    }
}
```

Figura 11 – Código em C# gerado pela IA ChatGPT ao ser perguntada como limitar as bordas

de uma câmera

Fonte: Autoria própria por meio da ferramenta ChatGPT (2023).

4.6.3 Temporizador e Coleta de Moedas

Considerando que um jogo onde o player pode apenas se movimentar e pular entre plataformas pode ser um pouco monótono, adicionei outras mecânicas para aumentar a dinamicidade e também o desafio em completar as diferentes fases. Assim como em outras partes do jogo, ChatGPT também gerou um passo a passo e um exemplo de código de como adicionar um temporizador que encerraria o jogo quando o tempo acabasse, e uma forma do player coletar moedas.

Com essa base, aprimorei o sistema para que a cada moeda coletada, além de acrescentar na UI o valor de 1 moeda, também acrescentasse 5 segundos no tempo restante para finalização da fase. E através do Midjourney, gerei uma imagem para a moeda que contivesse um ícone de ampulheta (Figura 12), para indicar que a moeda também tinha relação com o tempo da fase.



Figura 12 – Imagem gerada pela IA Midjourney, utilizando o *prompt: a golden coin icon with a hourglass icon in the center, in the style of nintencore, 2d game art, reduced palette, minimalist, white background --ar 1:1*

Fonte: Autoria própria por meio da ferramenta Midjourney (2023).

4.6.4 Contagem de Vida e Renascimento

Seguindo com a ideia de adicionar outras mecânicas para o jogo, também desenvolvi um sistema de vida para o jogador, onde o personagem possui no

máximo 3 vidas por fase, e ao perder a última vida, o jogador é apresentado à tela de Game Over. Junto dele, um sistema de *checkpoint*, que faz com que ao cruzar do personagem com uma placa específica, toda vez que ele perder uma vida após essa placa, ele retornará para a posição da placa e não à posição inicial da fase.

4.7 Interface do Usuário (*User Interface - UI*)

Todas as imagens usadas nas UIs, como botões e representações de vida e moedas, foram geradas através do Midjourney, necessitando apenas de um recorte do fundo da imagem, assim como em outras comentadas previamente.

Em alguns casos, como nos botões dos Menus e no temporizador das fases, foi necessário utilizar um elemento de texto posicionado em cima das imagens, visto que de acordo com os testes executados nesta pesquisa, a IA não consegue lidar bem com geração de imagens com palavras ou números, gerando imagens com palavras ou frases sem sentido e longe do que foi pedido, como demonstrado na Figura 13 em uma tentativa de gerar um logo para o jogo.



Figura 13 – Imagem gerada pela IA Midjourney, utilizando o *prompt*: *game art logo with text "Jungle Jumper", 2d art, pixel art*

Fonte: Autoria própria por meio da ferramenta Midjourney (2023).

4.8 Testes

Com o jogo finalizado, uma das partes mais importantes do desenvolvimento se inicia, a fase de testes. Nesta fase, é onde o desenvolvedor deve se colocar no lugar de um jogador comum e tentar “pensar fora da caixa”, e pensar em situações não tão convencionais, mas que são possíveis de serem reproduzidas dentro do jogo. É nela que vão ser identificadas as possíveis falhas no código do jogo, ou até mesmo nas animações ou em outras áreas.

Durante a execução desse experimento, um dos problemas que encontrei foi uma configuração dos colisores do personagem que constantemente estavam causando problemas na transição das animações. Após vários testes, consegui entender o que estava sendo causado, e apesar de não conseguir descrever o comportamento exato para que o ChatGPT tivesse uma conclusão direta para o problema, fiz alguns questionamentos que me ajudaram a chegar mais perto da solução.

4.9 Definição do Nome do Jogo

Com todos os problemas corrigidos, voltei ao ChatGPT para que pudessemos decidir o nome que daríamos pro jogo, afinal, o ChatGPT foi um co-desenvolvedor. Dentre algumas sugestões, como “*Lost in the Wild*”, “*Wilderness Escape*” e “*Jungle Safari Sprint*”, escolhi a primeira opção: “*Jungle Jumper*”.

4.10 Criação do Executável para Windows e Web

A última etapa da pesquisa foi criar uma versão do jogo que pudesse ser executada em computadores Windows, considerada a plataforma mais comum e acessível para jogos, e também em Web, facilitando o acesso para diversas pessoas em outras plataformas que possuem uma ferramenta com o Google Chrome instalada nos seus computadores. Assim como nas outras partes do desenvolvimento, o ChatGPT auxiliou com um passo a passo de todo o processo necessário para criar *builds* de cada uma das plataformas, e funcionou

perfeitamente conforme descrito.

4.11 Erros e Problemas Técnicos

Para alguns problemas encontrados, não foi possível contar apenas com o ChatGPT para resolver. Um dos exemplos foi citado anteriormente, onde o script gerado para a câmera seguir o personagem com um certo delay não estava funcionando, e precisei ir em busca das documentações da Unity para achar um método que correspondesse com o esperado.

Em adição, após implementar a mecânica da coleta de moedas, pensei em adicionar um efeito de reflexão de luz na imagem da moeda enquanto ela ainda não havia sido coletada, para que captasse a atenção do jogador. Para isso, foi necessário a criação de um Shader. O efeito causado pelo Shader gerado pelo ChatGPT inicialmente foi da moeda piscar em cores diferentes, o que seria longe do esperado. Mesmo assim, continuei tentando com diferentes *prompts*, mas não obtive sucesso. Depois de algumas tentativas, fui aconselhado pela IA a procurar um desenvolvedor especializado no assunto que pudesse me ajudar de forma mais assertiva:

“Peço desculpas pela dificuldade que você está enfrentando. A criação de Shaders altamente personalizados e complexos no Unity pode ser desafiadora. Se o Shader fornecido ainda não está atendendo às suas necessidades, você pode considerar a possibilidade de consultar um desenvolvedor especializado em Shaders ou um artista de efeitos visuais para criar o Shader personalizado de acordo com suas especificações.”

Por outro lado, durante o desenvolvimento notei no console da Unity alguns códigos de erro que não estavam muito claros sobre o que poderia ser, então recorri ao ChatGPT com a mensagem de erro recebida e fui prontamente respondido com a correção necessária. Além disso, também recebi respostas bem precisas com relação à perguntas técnicas sobre diferentes métodos nativos da Unity, como as diferenças entre Update, FixedUpdate e LateUpdate, e também qual a função do método Vector3.Lerp, por exemplo.

5 CONSIDERAÇÕES FINAIS

A partir do experimento realizado, foi possível avaliar, tal como previsto nos objetivos geral e específicos, diferentes contribuições e dificuldades para o uso de IA generativas para a criação de um jogo 2D. Contudo, apesar de nem todos os problemas terem sido resolvidos diretamente com o ChatGPT, e as imagens geradas através do Midjourney não estarem diretamente prontas para uso em um jogo, entendo que as ferramentas de IAs Generativas facilitaram muito o processo para um desenvolvedor individual com pouca experiência de uso da *game engine*, acelerando e encurtando o desenvolvimento em vários dias.

Existem muitas variáveis a serem consideradas antes de utilizar estas ferramentas profissionalmente, principalmente IAs que se classificam como geradoras de arte, como o Midjourney, pois muitas discussões ainda estão sendo feitas sobre a origem das imagens usadas para que essas IAs fossem treinadas, e quantas destas imagens possivelmente estariam cobertas por direitos autorais. De acordo James Vincent do periódico The Verge, em Janeiro de 2023, um trio de artistas dos Estados Unidos da América abriu um processo contra a Stability AI e Midjourney alegando que estas organizações infringiram os direitos de “milhões de artistas” ao treinar as suas ferramentas de IA em cinco bilhões de imagens extraídas da web “sem o consentimento dos artistas originais”. Atualmente, em Outubro de 2023, o processo ainda segue em andamento na corte da Califórnia.

Melhorias na área de IA Generativa vem sendo feitas diariamente, e novas ferramentas estão surgindo a todo momento. O ChatGPT já possui uma versão mais atualizada do que a usada nesta pesquisa, o ChatGPT-4, que conforme divulgado pela OpenAI, tem 82% menos probabilidade de responder a solicitações de conteúdo não permitido e 40% mais probabilidade de produzir respostas factuais do que o GPT-3.5, versão usada para esta pesquisa.

Finalmente, a partir da execução da pesquisa e do experimento, identificou-se para trabalhos futuros incluir a exploração de jogos em ambientes 3D ou de outros gêneros, bem como a utilização de diferentes *game engines* e novas ferramentas ou versões de IAs Generativa.

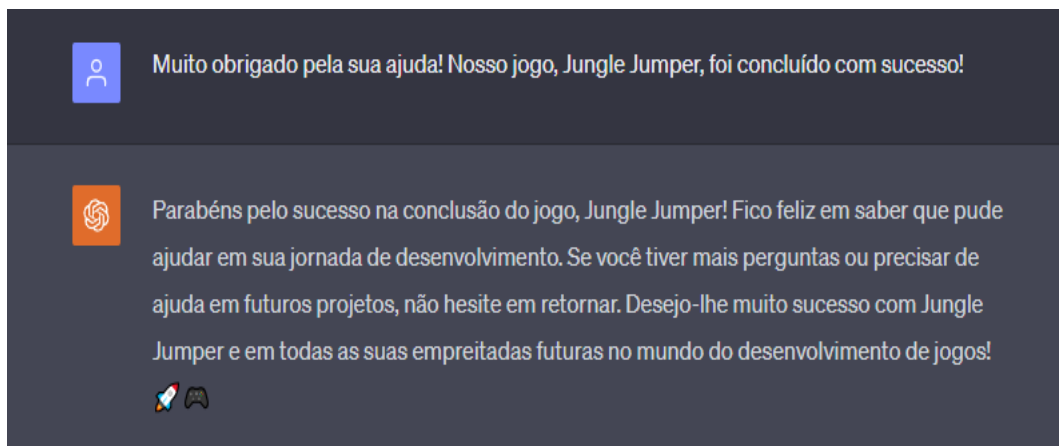


Figura 14 – Última interação com o ChatGPT antes da finalização da pesquisa

Fonte: Autoria própria (2023).

REFERÊNCIAS

Openai. **ChatGPT**. Disponível em: <https://chat.openai.com/>. Acesso em 04 de Novembro de 2023.

Midjourney. **Midjourney**. Disponível em: <https://www.midjourney.com/>. Acesso em 04 de Novembro de 2023.

French, Fiona; LEVI, David; MACZO, Csaba; SIMONAITYTE, Aiste; TRIANTAFYLLIDIS, Stefanos; VARDA, Gergo. **Creative Use of OpenAI in Education: Case Studies from Game Development**. Multimodal Technol. Interact. 2023, 7, 81. <https://doi.org/10.3390/mti7080081>

Beckwith, Michael. **ChatGPT Can Be Used for AI Generated NPC Dialogue and Voices in New Skyrim Mod**. Metro (Blog). 28 de Abril de 2023. Disponível em: <https://metro.co.uk/2023/04/28/chatgpt-can-be-used-for-ai-generated-npc-dialogue-in-new-skyrim-mod-18690766/>. Acesso em 12 de Outubro de 2023.

Surameery, Nigar M.S.; SHAKOR, Mohammed Y. **Use ChatGPT to Solve Programming Bugs**. Int. J. Inf. Technol. Comput. Eng. 2022, 3, 17–22. <https://doi.org/10.55529/ijitc.31.17.22>

Guana, Victor; STROULIA, Eleni; NGUYEN, Vina. "Building a Game Engine: A Tale of Modern Model-Driven Engineering," 2015 IEEE/ACM 4th

International Workshop on Games and Software Engineering, Florence, Italy, 2015, pp. 15-21, doi: 10.1109/GAS.2015.11

Nuclino. **Video Game Development Process: How successful games are made, from start to finish.** Disponível em: <https://www.nuclino.com/articles/video-game-development-process/>. Acesso em 12 de Outubro de 2023.

Freecodecamp. **What Is Game Development?** 28 de Dezembro de 2019. Disponível em: <https://www.freecodecamp.org/news/what-is-game-development/>. Acesso em 12 de Outubro de 2023.

Zenva. **What is Unity? – A Top Game Engine for Video Games.** Disponível em: <https://gamedevacademy.org/what-is-unity/>. Acesso em 12 de Outubro de 2023.

Deng Jianyang, & LIN, Yijia. 2022. **The benefits and challenges of ChatGPT: An overview.** *Frontiers in Computing and Intelligent Systems*, 2(2), 81-83. <https://doi.org/10.54097/fcis.v2i2.4465>

Gozalo-Brizuela, Roberto; GARRIDO-MERCHÁN, Eduardo C. 2023. **A survey of Generative AI Applications.** arXiv. arXiv:2306.02781

Lin, Lauren; LONG, Duri. 2023. **Generative AI Futures: A Speculative Design Exploration.** In *Proceedings of the 15th Conference on Creativity and Cognition (C&C '23)*. Association for Computing Machinery, New York, NY, USA, 380–383. <https://doi.org/10.1145/3591196.3596616>

Kaur, Davinder; SULEYMAN, Uslu; RITTICHER, Kaley J.; DURRESI, Arjan. 2022. **Trustworthy Artificial Intelligence: A Review.** *ACM Comput. Surv.* 55, 2, Article 39 (February 2023), 38 pages. <https://doi.org/10.1145/3491209>

Jovanović, Mlađan; CAMPBELL, Marc. **"Generative Artificial Intelligence: Trends and Prospects"** in *Computer*, vol. 55, no. 10, pp. 107-112, Oct. 2022, doi: 10.1109/MC.2022.3192720

Rozen, Riemer van. 2021. **Languages of Games and Play: A Systematic Mapping Study.** *ACM Comput. Surv.* 53, 6, Article 123 (November 2021), 37 pages. <https://doi.org/10.1145/3412843>

Philips, Cody; TRICK, Nico; NACKE, Lennart; MANDRYK, Regan. 2023. **The Role of Generative AI in Games Research.** In *Companion Proceedings of the Annual Symposium on Computer-Human Interaction in Play (CHI PLAY Companion '23)*, October 10--13, 2023, Stratford, ON, Canada. ACM, New York, NY, USA 2 Pages. <https://doi.org/10.1145/3573382.3616030>

Yadav, Ajay. **A Complete Comparison of ChatGPT, GPT-3, and GPT-4: What's the Real Difference?** Simplified, 21 de Maio de 2023. Disponível em: <https://simplified.com/blog/ai-writing/chatgpt-vs-gpt-3/>. Acesso em 13 de Outubro de 2023.

Vincent, James. **AI art tools Stable Diffusion and Midjourney targeted with copyright lawsuit**, 16 de Janeiro de 2023. Disponível em: <https://www.theverge.com/2023/1/16/23557098/generative-ai-art-copyright-legal-lawsuit-stable-diffusion-midjourney-deviantart>. Acesso em 11 de Novembro de 2023.

Brittain, Blake. **Judge pares down artists' AI copyright lawsuit against Midjourney, Stability AI**, 30 de Outubro de 2023. Disponível em: <https://www.reuters.com/legal/litigation/judge-pares-down-artists-ai-copyright-lawsuit-against-midjourney-stability-ai-2023-10-30/>. Acesso em 11 de Novembro de 2023.

Openai. **GPT-4 is OpenAI's most advanced system, producing safer and more useful responses**, 2023. Disponível em: <https://openai.com/gpt-4>. Acesso em 12 de Novembro de 2023.

Ramadan, Rido; WIDYANI, Yani. **Game development life cycle guidelines**. Proceedings of the 2013 International Conference on Advanced Computer Science and Information Systems (ICACISIS), pp. 95-100. IEEE, 2013. <https://jiharu.github.io/lowpoly/img/06761558GameDev.pdf>

Manuscrito recebido em: 03 de Dez. de 2023

Aprovado: 26 de Dez de 2023

Publicado: 29 de Dez de 2023