

# Crônicas de Caim: Uma aventura Top Down em Pixel Art

Guilherme de Paula Ferreira<sup>1</sup>, Gabriel C. Dall’Pozzo<sup>1</sup>

<sup>1</sup>Centro Universitário Campo Real

R. Comendador Norberto, 1299 - Santa Cruz - 85015-240

Guarapuava – PR – Brazil

{engs-guilhermeferreira, prof\_gabrielpozzo}@camporeal.edu.br

**Abstract.** *Digital games have been gaining more space in people's routines every year, spanning across all age groups from children to the elderly. This project aims to create and develop a game focused on the digital gaming field. A 2D game in a top-down style with Pixel Art animations was created, named "Chronicles of Caim." It is a Role-Playing Game (RPG) designed for Windows, with plans for expansion to mobile devices running the Android operating system. To carry out the project, a systematic mapping research methodology was adopted, combined with the agile Scrum development methodology. The use of these two approaches aimed to provide efficiency and yield positive results for the completion of this project. Consequently, a demonstration (Demo) of the proposed game was developed, featuring a menu, pixel art graphics, a dialogue system, environmental sound effects, and the implementation of all mechanics, ensuring their functionality and delivering a complete gaming experience.*

**Keywords:** *Game Development, 2D Games, Pixel Art, Top Down.*

**Resumo.** *Jogos digitais vêm ganhando um espaço maior a cada ano na rotina das pessoas de todas as idades, de crianças à idosos, este trabalho tem o objetivo de criação e desenvolvimento de um jogo, focado na área de jogos digitais, foi desenvolvido um jogo em 2D no estilo top down e com animações em Pixel Art, nomeado como “Crônicas de Caim”, um jogo de Role Playing Game (RPG) para windows, com planos de expansão para dispositivos móveis com o sistema operacional Android. Para a realização do projeto foi adotado a metodologia de pesquisa do mapeamento sistemático combinado com o desenvolvimento ágil Scrum como metodologia de desenvolvimento. A utilização destes dois critérios teve como finalidade proporcionar eficiência e trazer bons resultados para a conclusão deste projeto. Sendo assim foi desenvolvido uma demonstração (Demo) do jogo proposto com menu, artes desenvolvidas em pixel art, sistema de diálogos, sonorização do ambiente e o desenvolvimento de todas as mecânicas deixando-as funcionais assim trazendo uma experiência completa do jogo.*

**Palavras-Chave:** *Desenvolvimento de Jogos, Jogos 2D, Pixel Art, Top Down.*

## 1. INTRODUÇÃO

A indústria de jogos, desde sua origem em meados de 1980, tem testemunhado uma evolução notável, tanto em termos econômicos quanto tecnológicos, desempenhando um papel fundamental na economia global, gerando bilhões de dólares anualmente por meio de uma ampla variedade de formatos de jogos. Essa significativa contribuição financeira, destaca o impacto substancial que a indústria dos jogos tem no cenário econômico mundial. Somente no Brasil, o mercado de “*games*” teria ultrapassado, em 2021, cerca de USD 2,3 Bilhões, o que posicionaria o 10º no mundo (ITAMARATY, 2022). Até 2024, mundialmente, irá ultrapassar os US\$ 220 bilhões (FORBES, 2023).

O estudo sobre jogos digitais pode ser entendido de maneira mais simplificada através do ponto de vista do *game design*, no qual reúne diversos conteúdos para criação de jogos, como artes visuais e sonoras, programação e produção, mecânicas de jogos, narrativas, entre outros (LUCENA, 2014). É o responsável por todo o conceito do jogo, sendo descrito características principais como personagens, forma do jogo, ou seja, todos os elementos que fazem parte do projeto (LEITE, MENDONÇA, 2013).

Diante de todo esse contexto foi desenvolvido uma demo do jogo *Crônicas de Caim: Uma Aventura Top Down em Pixel Art*, o desenvolvimento do produto descrito neste trabalho acontece devido a familiarização ao mundo dos jogos digitais. Sendo um jogo desenvolvido em 2D no estilo *top down*, e possuindo a perspectiva ou visualização em duas dimensões em que a cena ou objeto é observado de cima, em sua maioria para dispositivos móveis ou os famosos *Role playing games* (RPG's) encontrados na década de 90 (FEIJÓ e DIEGO, 2020).

O projeto enfrenta desafios como a familiarização com ferramentas incluindo a *Engine Unity* que abrange tanto jogos em 3D (jogo tridimensional) como 2D (jogo bidimensional) o mesmo conta com suporte a linguagens como C#, C++ e Java. O jogo “Caim” também irá prever a interação entre o protagonista e *Non-playable-characters* (NPC's) que são personagens não controláveis pelo jogador, estes existem tanto em jogos *multiplayer*, vários jogadores, quanto *singleplayer*, um jogador, sendo que neste há um maior destaque. (BORGES et al, 2009).

Com o desenvolvimento do jogo descrito, espera-se que desperte grande interesse tanto entre os jogadores em potencial quanto em outros tipos de público, devido à sua disponibilidade em plataformas que alcançam uma ampla parcela de usuários. Além disso, o projeto também visa despertar o interesse dos futuros acadêmicos para desenvolver projetos similares.

### 1.1 OBJETIVOS

Objetivou-se o desenvolvimento de uma demo para o jogo “*Crônicas de Caim*” sendo este elaborado no estilo em *Top Down* e a estilização em *Pixel Art*, técnica de produzir imagens digitais sendo utilizada desde seu auge, em meados de 1990, quando consoles e computadores tinham muitas limitações (ALENCAR, 2017). O projeto deverá proporcionar entretenimento e diversão aos jogadores, permitindo que as pessoas mergulhem em um mundo virtual e exploratório desfrutando de desafios interativos. O jogo busca envolver os jogadores em uma experiência imersiva, à qual eles poderão se conectar com a história do mundo e dos personagens que neste habitam.

Dentro desse contexto, os objetivos específicos do trabalho foram fundamentais para impulsionar a conclusão do projeto sendo:

- Desenvolver uma história envolvente e rica, por meio de diálogos bem elaborados e eventos significativos, que transportem os jogadores para um mundo fictício repleto de aventuras, mistérios e reviravoltas e construir um enredo cuidadosamente elaborado para despertar o interesse dos jogadores e incentivá-los a explorar cada vez mais o vasto universo do jogo.
- Proporcionar aos jogadores uma facilidade de aprendizagem ou o desenvolvimento de habilidades específicas, a fim de auxiliá-los na resolução de problemas.
- A oportunidade de desenvolver um software desde o início e proporcionar aprendizados de conceitos relacionados a área de jogos digitais, como o uso de *engine*, *game design*, *level design*, tipos de mecânicas, entre outros.

## 1.2 ORGANIZAÇÃO DO DESENVOLVIMENTO DO TRABALHO

Este trabalho está organizado em 6 capítulos. No Capítulo 1, foi apresentada a introdução, objetivo geral e específicos. No capítulo 2, será apresentado o referencial teórico. No capítulo 3, o estado da arte e a metodologia de pesquisa abordada. No capítulo 4, foi descrito sobre a metodologia de desenvolvimento utilizada. No capítulo 5, é exibido os resultados do projeto e sua programação. Por fim, no capítulo 6 será apresentado sobre a conclusão do trabalho.

## 2. REFERENCIAL TEÓRICO

Este Capítulo apresenta o referencial teórico da pesquisa, trazendo os conceitos teóricos deste trabalho. A seção 2.1 explica sobre o mundo dos jogos digitais. A seção 2.2 apresenta sobre o papel do *Game Designer* para o desenvolvimento e criação de um jogo.

### 2.1 JOGOS DIGITAIS

Os jogos digitais possuem uma história muito rica e evolutiva no qual abrange várias épocas, desde os primórdios de 1970 e 1980 até a atualidade. Antes da era eletrônica, jogar era uma atividade ancestral e inerente ao entretenimento humano, predominantemente voltada ao lazer, diante disto a atribuição do *status* de jogo pode ser condicionada pela habilidade de um indivíduo identificar-se como participante, seguindo as suas regras e explorando os benefícios relativos a essa experiência. Em um segundo plano, esta classificação também depende da interpretação dos observadores, que tendem a buscar e compreender a natureza da atividade em questão, assim determinando se ela desperta curiosidade ou proporciona entretenimento. Apenas após criar-se compatibilidade entre estas diferentes perspectivas é que será possível dividir a estrutura sistêmica ao conceito jogo (GULARTE, 2010).

Tanto os jogos digitais quanto os “tradicionais”, como as brincadeiras de pega-pega e amarelinha, espelham o contexto, social, cultural, histórico e econômico onde estão inseridos. A indústria de jogos digitais espelha a paisagem contemporânea da indústria cultural (DANTAS, 2012b, p. 220). A partir da década de 70, a cada ano que se passou as evoluções tecnológicas estão evidentes. A partir da quarta geração, os

videogames eram capazes de reproduzir gráficos de 16-bits e 32-bits muito mais detalhados, coloridos e com um processamento e velocidade superior (HARRIS, 2014). Desde então, os jogos eletrônicos começaram a ficar populares, desde a época dos arcades até as novas gerações encontradas na atualidade.

Os jogos estão muito além do entretenimento podendo estar presentes na educação (AZEVEDO, 2012), pois a partir do grande avanço tecnológico do século 21, o acesso aos jogos eletrônicos dentro e fora da sala de aula está mais fácil, como também para influenciar e aprimorar a cognição, trazendo benefícios para os usuários independente da idade, pois auxilia no raciocínio lógico, memória, percepção e habilidades especiais (BANIQUEDE et al, 2014).

## 2.2 GAME DESIGN

*Game Design* engloba todos os aspectos de uma *gameplay* de um jogo, sendo composto pelo ambiente, os personagens, as mecânicas, a narrativa, músicas e sons. Jogos digitais são criados por *designers* ou equipes de desenvolvedores sendo consumidos por jogadores. O *Game Design* tem como principal papel organizar todos estes aspectos tendo como objetivo que eles não fiquem presos uns com os outros. O *design* e o jogador possuem perspectivas diferentes. É importante considerar essas perspectivas, pois qualquer pequena mudança em uma camada pode afetar outras em cascata. Pensar no jogador irá incentivar o *design* trazendo assim uma melhor experiência para o jogador (HUNICKE, LEBLANC, 2004).

Cada jogo é único, o *game designer* deve aproveitar a expectativa do consumidor, inovando e trazendo funcionalidades diferentes das esperadas pelos consumidores de jogos digitais (LEMES, 2009). A figura do *game designer* é uma das grandes mentes por trás de um jogo, responsável por toda a criação, planejamento e construção de todos os conceitos criativos e ambientação a qual o usuário está inserido (LUCENA, 2014). Para isso, é fundamental que o desenvolvedor faça uso de características psicológicas bem delimitadas, com base em peculiaridades como, enredo, jogabilidade, objetivos, fases, *interfaces* e experiência do usuário (LEMES, 2009).

Para a realização de qualquer desenvolvimento é necessário estar dentro de alguma metodologia de *design*, seja por protótipo físico, construção de argumento ou por meio de implementações de experimentos controlados. O *game design* desempenha o papel de organizar o projeto ao todo, e podendo ter a ajuda do time inteiro de desenvolvimento, pois conforme é realizado a idealização do jogo, já está sendo criado o *design* do mesmo. Apesar de um jogo ser desenvolvido com um objetivo em específico, os jogadores serão os que irão modificá-los, ou seja, o *game designer* irá criar as mecânicas e o jogador irá buscar um jeito para usá-las da melhor forma. Os jogadores também poderão descobrir coisas que os desenvolvedores nem sequer pensaram assim quebrando o jogo (HUNICKE, LEBLANC, 2004).

O campo do *game design* é uma área de pesquisa e desenvolvimento que desafia constantemente os limites da criatividade e da interação humana com a tecnologia. Em relação ao mundo dos jogos digitais, percebe-se que por trás de cada experiência interativa, há um *game designer* dedicado, cujo trabalho é aprimorar as mecânicas de jogo para criar uma experiência atraente aos jogadores. Mecânicas em jogos costumam ser relativamente parecidas, porém há inovações em pontos importantes, como a utilização de magias, acessórios, itens e também na utilização de cenários, além

surgirem novos destes ao decorrer do jogo, altera-se a chamada jogabilidade ou o desenvolvimento do jogador durante uma partida (LUCENA, 2014).

### 3. ESTADO DA ARTE

Os jogos digitais são definidos como um sistema onde jogadores são envolvidos em um mundo com desafios abstratos, regidos por regras e interatividade aos quais por sua vez, trazem respostas emocionais, ou seja, oferecem ao jogador diversão, motivação e imersão, tornando a experiência do jogo mais rica e significativa (KOSTER, 2013).

Por este capítulo será apresentado um mapeamento sistemático sobre o estudo e desenvolvimento de jogos eletrônicos. Os objetivos do estudo serão principalmente analisar jogos, mecânicas e ferramentas utilizadas ou desenvolvidas de maneira independente, com o propósito de promover o desenvolvimento pessoal, acadêmico ou profissional.

As principais contribuições do estudo serão levantar dados relacionados a criação de jogos eletrônicos para o desenvolvimento do jogo proposto neste artigo. A seção 3.1 será explicado sobre a metodologia do mapeamento sistemático. Na seção 3.2 mostrará os resultados da pesquisa e discussões. A seção 3.3 será discutida sobre os trabalhos relacionados ao tema do projeto. Na seção 3.4 será apresentado um comparativo entre os jogos. A seção 3.5 serão descritas as considerações finais do capítulo.

#### 3.1 METODOLOGIA DE PESQUISA

Os estudos e o desenvolvimento de jogos digitais e seus elementos estão sendo cada vez mais realizados e aumentando em ritmo acelerado. Em grande parte dos casos, a metodologia de desenvolvimento de jogos é pouco definida e, frequentemente, não segue um processo estruturado, sendo muitas vezes baseada em tentativa e erro (DE CASTRO, 2023). Com o intuito de preencher tal lacuna foi realizado o mapeamento sistemático de pesquisas existentes, sobre o tema envolvendo a criação e desenvolvimento de jogos.

O mapeamento sistemático é uma metodologia no qual compreende uma pesquisa criteriosa na literatura, tendo como o maior objetivo avaliar a abrangência, o escopo e o volume de estudos publicados, geralmente referidos como estudos primários, dentro de um domínio específico de interesse (PETERSON et al., 2008).

Este mapeamento pode ser classificado como estudo secundário quando tem como objetivo fornecer uma visão abrangente de uma área de pesquisa por meio da análise, quantificação e identificação de resultados de estudos primários. Este método pode ser indicado a ser usado quando há uma escassez de estudos primários sobre o tema de interesse ou quando a área de pesquisa é extensa (GALVÃO, 2020).

A elaboração do mapeamento foi orientada pelo método descrito por Petersen et al. (2008), no qual compreende cinco etapas sendo: (i) formulação de questões de pesquisa, (ii) construção das *strings* de busca, (iii) Análise dos artigos retornados, (iv) classificação dos artigos, e (v) a extração dos dados.

Foi definido o período de busca por artigos publicados desde 2021 para realizar as análises mais recentes sobre o assunto, tendo em vista que os primeiros trabalhos publicados ocorreram em 1999.

### 3.1.1 Questões de Pesquisa

As questões de pesquisa (QP) consideradas neste mapeamento serão:

- QP1: Quais os tipos de mecânicas similares podem ser utilizadas ?
- QP2: Quanto a estilização, por que utilizar o *pixel art* e como ele pode agregar valor no produto final ?
- QP3: Por que utilizar a perspectiva de câmera de jogo no estilo *Top Down*?
- QP4: Por que realizar animações na plataforma da *Engine Unity* ?

### 3.1.2 Condução de Busca

O repositório de busca escolhido foi o *google*<sup>1</sup> acadêmico que é uma ferramenta gratuita que tem a função de localizar trabalhos acadêmicos de vários tipos como por exemplo, artigos de congressos, teses e dissertações, em múltiplas línguas, podendo determinar a frequência em que foram citados por outras publicações acadêmicas (CAREGNATO, 2011) .

O Quadro 1 demonstrará as expressões que foram utilizadas por meio deste mapeamento. A *String* foi implementada levando em conta as palavras-chaves principais relacionadas ao tema do trabalho, trazidas através de artigos científicos.

Quadro 1 - Strings de Busca

String	Expressão
1	("Mecânica" AND "Pixel" AND "Top Down" OR "Animação" OR "Unity").

Fonte: O autor

O mapeamento foi realizado por um pesquisador juntamente com as questões de busca e *strings*.

### 3.1.3 Análise dos Artigos

Para a análise e seleção dos artigos, foi estabelecido critérios de inclusão (CI) sendo:

- CI1: Trabalhos publicados no ano de 2021.
- CI2: Artigos na língua portuguesa.
- CI3: Artigos completos.

Os artigos que atenderem esses critérios, serão reavaliados sobre os critérios de exclusão (CE), trabalhos nos quais não atendam algum destes critérios são retirados do mapeamento. Sendo assim os critérios de exclusão adotados, foram:

- CE1: Artigos que não falem sobre criação de jogos.
- CE2: Artigos duplicados.
- CE3: Artigos que possam ser visualizados de forma integral.

### 3.1.4 Classificação e Extração dos Dados

Ao utilizar a *String* de busca foi possível buscar 1560 referências fornecidas no repositório do *google*<sup>1</sup>acadêmico para o mapeamento. Após efetuar os critérios de inclusão e exclusão estipulados anteriormente que convergiam com a pesquisa, então foram selecionados 8 artigos, estes sendo comentados abaixo e 3 destes farão parte dos trabalhos similares do trabalho, possuindo similaridades ao projeto proposto para desenvolvimento. O mapeamento sistemático ajudou a trazer ideias de outros trabalhos já desenvolvidos para adaptar ao jogo proposto neste trabalho.

### 3.2 RESULTADOS E DISCUSSÕES

Seguindo metodologia do mapeamento sistemático e realizando as suas etapas foi possível inicialmente o retorno de um total de 263 artigos utilizando os critérios de inclusão, na sequência foi realizado uma leitura rápida e aplicado os critérios de exclusão, desconsiderando artigos que não falem sobre a criação de jogos, artigos duplicados ou artigos que não pudessem ser visualizados. Dando continuidade, após analisado e avaliado o título, caso o mesmo não tivesse relação ao tema abordado, seria retirado, ao final deste processo, permaneceu 19 documentos. Para a próxima etapa, houve a abertura de cada documento e leitura do resumo restando 12 artigos. Na sequência ocorreu a avaliação de cada um destes, por meio de uma leitura completa dos principais pontos abordados, sendo avaliado similaridades ao jogo no qual será desenvolvido, durante esse processo de leitura restou um total de 8 artigos para a continuação do estudo e discussão.

A primeira publicação citada neste mapeamento faz parte de um trabalho de conclusão, De Castro (2023) desenvolveu de forma simplificada um cenário de jogo que lembra uma instituição de ensino, desenvolvido no *Unity*, em *pixel art* e no estilo *top down*, com mecânicas simples do personagem utilizando as teclas "W", "A", "S", "D" ou as setas direcionais podendo dialogar ao se aproximar de NPCs.

Adão (2023), desenvolveu um projeto de *concept art* para um futuro jogo eletrônico tendo como objetivo o desenvolvimento visual de personagens, trazendo um roteiro em lendas e cultura afro-brasileira, sendo um jogo de aventura em 2D possuindo mecânicas diversas podendo mover utilizando as teclas W", "A", "S", "D" ou as setas direcionais, esquiva, sistema de pontos, inventário e a possibilidade de poder interagir com NPCs.

Santos (2023), realizou uma análise por meio do desenvolvimento de um jogo no qual foi utilizado uma interface minimalista com gráficos em 2D e no estilo *pixel art*, utilizando a engine *Unity* com uma mecânica *point-and-click* que utiliza o uso do mouse e também conta com um sistema de diálogo, o jogo foi postado na plataforma *Itch.io*.

O trabalho desenvolvido por Chies (2023), aborda formas de agradar diversas personalidades de jogadores por meio de mecânicas e elementos de *game design* por meio de jogos competitivos. O jogo desenvolvido foi um *single player* no formato de um *action shooter* no qual a cada fase há um novo inimigo para confrontar. Utiliza de gráficos em *pixel art* e no estilo *top down*, tendo mecânicas para atirar contra o inimigo e de movimentação simples utilizando as teclas W", "A", "S", "D" do teclado podendo

---

<sup>1</sup> <https://scholar.google.com.br/?hl=pt>

mover-se em 8 direções sem opções adicionais de correr ou pular. O projeto foi utilizado a *engine Unity* para desenvolvimento.

No trabalho desenvolvido com o estilo em *pixel art* por Oliveira (2023), explora a implementação da animação, destacando limitações e vantagens deste estilo visual. A narrativa e caracterização dos personagens fazem parte da arte e comunicação visual. Um fator de grande importância no jogo é a leitura rápida da barra de vida, no qual se acaba rapidamente caso um inimigo acabe encostando no personagem.

Conforme foi analisado nos artigos destacam-se: algumas mecânicas similares para a movimentação do personagem, o estilo do jogo em *top down*, a ferramenta de desenvolvimento para a criação *Unity* e quase em todos os artigos analisados a estilização optada foi em *pixel art*.

Entre os artigos selecionados, 80% deles sugerem trabalhos futuros, sendo que um destes não havia ainda programação, porém toda a ideia formada para desenvolvimento, um dos trabalhos foi publicado, finalizado e disponibilizado para *download* na plataforma online *itch.io*, e o outro foi liberado uma demonstração com o código fonte via Google Drive.

### 3.2.1 Respostas às Questões de Pesquisa

Esta seção mostrará as respostas das questões de pesquisa definidas no início do mapeamento.

A. **QP1** : Quais os tipos de mecânicas similares podem ser utilizadas ?

Conforme Kirchof (2023), jogos que envolvem cenários podem ter interfaces similares e funcionalidades de mecânicas similares, por meio dessa identificação pode-se concluir que os jogos independentemente do gênero proposto podem acabar ocorrendo de se repetirem, auxiliando no processo de adaptação e controle com o jogador.

B. **QP2** : Quanto a estilização, por que utilizar o *pixel art* e como ele pode agregar valor no produto final ?

Os gráficos pixelados ocupam uma boa parte do mercado mundial de jogos, não somente dos jogos “retrô” vindos dos clássicos antigos, mas também dos jogos que utilizam o *pixel art*, no qual possui um estilo próprio de arte em elementos gráficos, lançados anualmente principalmente por empresas *indies* (Silva, 2023).

C. **QP3** : Por que utilizar a perspectiva de câmera de jogo no estilo *Top Down*?

Segundo De Castro (2023) jogos que são desenvolvidos com uma perspectiva de câmera de jogo no estilo *top down* oferecem uma melhor visão de cena. As artes em *pixel art*, juntando ao estilo em *top down* transmitem um ambiente “retrô” e ao mesmo tempo um ambiente bonito, charmoso e leve, trazendo ao jogador uma forma de jogar totalmente diferente do estilo em 3D.

D. **QP4** : Por que realizar animações na plataforma da *engine Unity* ?

Conforme Guerreiro (2021) a utilização do *Unity* possui uma alta capacidade de desenvolvimento de jogos em 2D, possuindo interfaces simples e fáceis de se aprender, assim como é possível utilizar de técnicas de animação utilizando a plataforma, facilitando o processo sem precisar usar *softwares* de terceiro, foi observado que dentre os artigos localizados em sua grande maioria utilizava a própria *engine* para a criação das animações.



### 3.3 TRABALHOS RELACIONADOS

Os jogos que serão descritos a seguir possuem similaridades ou algum outro aspecto ao jogo que será desenvolvido descrito neste artigo.

#### 3.3.1 Dr Solinho

O trabalho desenvolvido por (GUIMARÃES et al., 2022), tem o objetivo de trazer um jogo educacional com características de RPG e visualização *top down* e com artes em *pixel art*. O jogo foi baseado em jogos “retrôs” consagrados em seu gênero. Quanto às mecânicas, o jogo se passa em um mundo aberto no qual o personagem pode andar livremente e interagir com os personagens em sua volta. É possível realizar algumas ações como plantar, regar e coletar água. Também é possível realizar combates em turnos, o objetivo do jogo é realizar diversas tarefas apresentadas durante sua *gameplay* para que assim os objetivos sejam alcançados.

#### 3.3.2 Two Kingdoms

*Two Kingdoms* é um outro jogo no qual se baseia no mundo chamado “Soulwin”, no estilo *action* RPG em 2D com uma câmera no estilo *top down* e artes em *pixel art*. O objetivo do jogo é junto com um jovem chamado “Agnes” restabelecer a paz de seu mundo, derrotando inimigos e superando os desafios que aparecem em seu caminho (GUERREIRO et al., 2021).

#### 3.3.3 Jogo sério para auxílio da educação ambiental

Este trabalho desenvolvido por (KIRCHHOF et.al., 2023), trás o conceito de um simulador de fazenda com foco em sustentabilidade possuindo escolhas livres por meio de cliques na tela, um jogo desenvolvido com artes em 2D e com estilo em *pixel art*, o jogo possui um sistema de *feedback* em tempo real das ações do jogador permitindo que o jogador perceba a diferença na forma que gerencia a sua fazenda e possa ser motivado a jogar, segundo o autor é a partir do *feedback* que um jogo de aprendizagem acontece.

### 3.4 COMPARATIVO DE JOGOS

No Quadro 2, pode ser observado a comparação entre jogos que possuem características similares ao jogo que será desenvolvido. O jogo será baseado utilizando artes em *pixel art* e 2D, com estilo *top down*, sendo possível interagir com NPC's. Para o desenvolvimento de “Crônicas de Caim” será adaptado e desenvolvido utilizando todos os recursos mostrados no quadro abaixo.

Quadro 2 - Comparativo de Semelhanças e Recursos Disponíveis entre os jogos.

RECURSO	DR. SOLINHO	TWO KINGDOMS	JOGO SÉRIO PARA AUX. EDUCAÇÃO	CRÔNICAS DE CAIM
Pixel art 2D	Possui	Possui	Possui	Possui
Top Down	Possui	Possui	Não Possui	Possui
História envolvendo NPC's	Possui	Possui	Não Possui	Possui
Mecânica de Pesca	Não Possui	Não Possui	Não Possui	Possui
Mecânica de Construção	Não Possui	Não Possui	Não Possui	Possui
Mecânica de Combate	Possui	Possui	Não Possui	Possui
Mecânica de Plantação	Possui	Não Possui	Possui	Possui

Fonte: O autor

### 3.5 CONSIDERAÇÕES FINAIS DO CAPÍTULO

Este capítulo relata sobre a metodologia de pesquisa utilizada a qual ajudou a trazer artigos com jogos em desenvolvimento ou já desenvolvidos através de trabalhos acadêmicos de uma maneira mais fácil e prática nos quais possuem técnicas ou alguma outra similaridade ao jogo proposto neste artigo.

## 4. DESENVOLVIMENTO DO JOGO CRÔNICAS DE CAIM: UMA AVENTURA TOP DOWN E EM PIXEL ART

Neste capítulo será apresentado sobre o desenvolvimento do jogo “Crônicas de Caim”. A seção 4.1 mostrará a metodologia de pesquisa e de desenvolvimento. Na seção 4.2 será discriminado sobre as ferramentas utilizadas. A seção 4.3 será apresentada sobre os sites com recursos. A seção 4.4 será apresentada sobre os componentes do jogo.

### 4.1 METODOLOGIA DE PESQUISA E DE DESENVOLVIMENTO DO JOGO CRÔNICAS DE CAIM

Para a realização do desenvolvimento deste trabalho o mesmo foi dividido em duas partes sendo a primeira, a explicação sobre a metodologia de pesquisa e a segunda sobre a metodologia para o desenvolvimento do jogo proposto.

#### 4.1.1 Metodologia de pesquisa

Para o desenvolvimento da pesquisa foi necessário realizar primeiramente a definição do tema do projeto, ou seja, a criação e desenvolvimento de um jogo 2D em *top down* no gênero RPG utilizando artes em *pixel art*.

Após a definição do tema foram levantados trabalhos no qual tenham relação à criação e desenvolvimento de jogos digitais. Decidiu-se utilizar o mapeamento sistemático a fim de trazer trabalhos onde haja algo em comum sendo a estilização, mecânicas, podendo ser ou sendo em 2D (duas dimensões) e também desenvolvidos na plataforma de desenvolvimento do *Unity*.

Após a execução do mapeamento sistemático foi separado e analisado os exemplos de trabalhos referentes à criação e desenvolvimento de jogos com similaridades no qual irá compor o jogo desenvolvido “Crônicas de Caim”.

#### **4.1.2 Metodologia de Desenvolvimento do Produto**

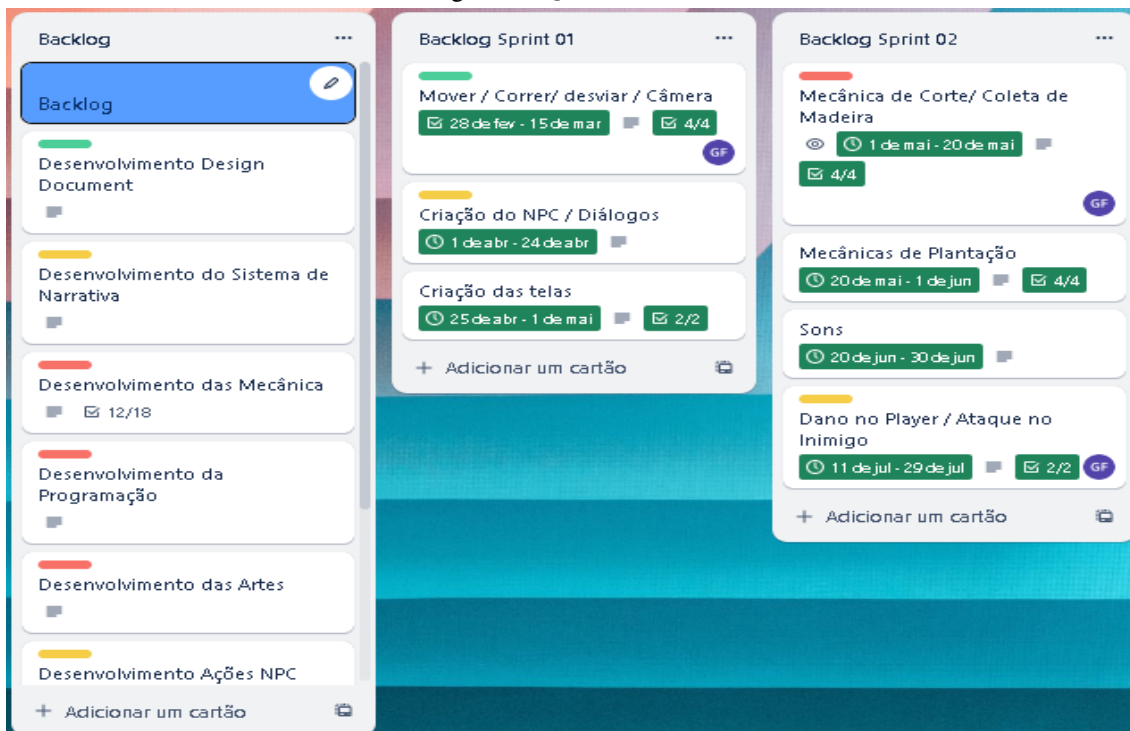
Durante o processo de organização de todas as funcionalidades do jogo “Crônicas de Caim”, adotou-se a utilização da metodologia de desenvolvimento ágil *Scrum*. O conceito das metodologias ágeis emergiu em meados dos anos 2000, principalmente no cenário da Ciência da Computação (DOS SANTOS, 2004). O *Scrum* é bem documentado para o desenvolvimento de software e aplicações digitais, tendo a sua aplicação original voltada para a elaboração de software ou projetos de qualquer tamanho ou complexidade (SUTHERLAND, 2014).

No contexto do desenvolvimento do jogo “Crônicas de Caim”, a metodologia *scrum* foi escolhida para melhorar a eficiência e o gerenciamento das atividades que serão desenvolvidas, permitindo a adaptação de mudanças no processo de criação. Foram criadas as atividades das quais serão necessárias para realizar a finalização do processo, foi levantado as respectivas prioridades e complexidades e com um prazo para o desenvolvimento, tendo como objetivo poder completar o projeto de forma rápida e eficiente.

Para a realização e aplicação da metodologia foi utilizado a plataforma *trello* que é um aplicativo gratuito para gerenciamento de projetos *online*, servindo tanto para organização de tarefas individuais, quanto para equipes, possuindo um visual limpo e estruturado (TRELLO, 2023). Baseando-se na metodologia *kanban*, os projetos são representados por quadrados, e através dele são atribuídas tarefas através de cartões, sendo possível definir prazos, inserir comentários, criar *checklist* dentre várias outras opções, assim possibilitando o gerenciamento de etapas pelos demais membros da equipe.

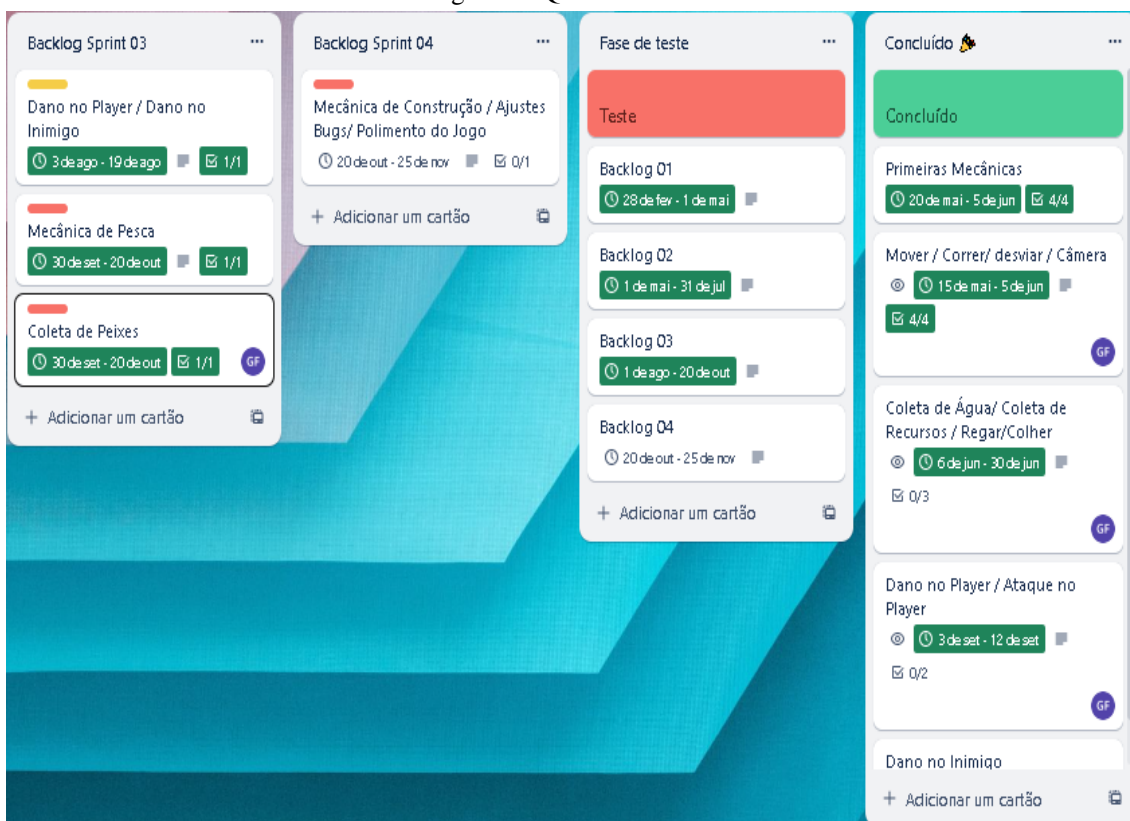
Na Figura 1 e 2, inicialmente foram definidas as tarefas pendentes no quadro *backlog*, foi definido três cores conforme a prioridade e dificuldade sendo, verde como uma tarefa fácil, amarelo moderado e vermelho difícil, na sequência foi definido *sprints* com prazos variando entre 1 e 3 semanas para cada tarefa a ser desenvolvida, foi definido um quadro como concluído no qual irá conter as tarefas finalizadas.

Figura 1 - Quadro Kanban



Fonte: Criação do Autor

Figura 2 - Quadro Kanban



Fonte: Criação do Autor

## 4.2 FERRAMENTAS

Para a realização do produto em questão foi utilizado ferramentas para o seu desenvolvimento e para a criação e modificação de artes visuais em *pixel art*. A seguir, será apresentado as ferramentas utilizadas durante a execução do projeto.

### 4.2.1 Unity

A *Unity* é uma ferramenta para o desenvolvimento de jogos desenvolvida pela *Unity Technologies* e lançada em Junho de 2005 (HUSSAIN et al., 2020). Possui um acesso fácil e intuitivo, contendo uma documentação robusta ajudando no aprendizado de desenvolvedores iniciantes. Para utilizar a sua aplicação, conforme *Unity* (2020) é fornecido componentes de física e renderização ajudando e reduzindo a complexidade do projeto durante o seu desenvolvimento.

Uma *engine* gráfica possui diversos recursos, utilizados para criar um ambiente bonito e agradável, sendo assim um pilar para o desenvolvimento de jogos digitais, com ele é possível criar modelos 2D, 3D, VFX (efeitos visuais), texturas, áudio, GUI (Interface gráfica), entre outros (ACTON, 2014).

A *Engine Unity* possui uma interface simples e muito amigável, facilitando o desenvolvimento de jogos digitais de diferentes gêneros e plataformas (PASSOS, 2009).

### 4.2.2. Microsoft Visual Studio

O *Microsoft Visual Studio* é utilizado para desenvolver qualquer tipo de aplicativo, por ele é possível trabalhar no ambiente de desenvolvimento integrado (IDE) do *Visual Studio*. Possui suporte para C++, C#, Java, Python, PHP entre outros. Pode ser utilizado em plataformas que contam com execuções dinâmicas, rodando em tempo de execução, utilizada principalmente em plataformas .NET e *Unity* (MICROSOFT, 2023).

### 4.2.3. Pixilart

O *Pixilart* é uma ferramenta gratuita que pode ser acessada através do site [pixilart.com](https://pixilart.com), este possui uma interface interativa e atrativa para o usuário, é uma plataforma social para realizar a criação de *pixel art*, sendo considerado um novo tipo de rede social com o objetivo de trazer engajamento na arte de apreender e fazer *networking*. A ferramenta incentiva a programação e a arte digital (PIXILART, 2023).

### 4.2.4. Aseprite

O *software Aseprite* é um *software* pago desenvolvido pela *Igara Studio SA*, atualmente encontra-se em seu site oficial por US\$ 19,99, sendo possível comprar diretamente pelo site da plataforma da Steam que é um software de gestão de direitos digitais criado pela *Valve Corporation*, trazendo jogos e aplicativos de programação, fornecendo-os a preços acessíveis (STEAM, 2023). Através do *Aseprite* é possível criar animações em 2D no qual podem ser usados em qualquer plataforma, sendo possível criar de *sprites* a *pixel art*, gráficos em estilo “retrô” e tudo o que precisar criar na era de 8 ou 16 bits (ASEPRITE, 2023).

### 4.3 SITES COM RECURSOS

A seguir será explicado sobre os sites utilizados, contendo diversos recursos para auxiliar no desenvolvimento do projeto.

#### 4.3.1 Itch.io

O site *ITCH.io* é uma maneira fácil de encontrar e compartilhar jogos independentes (*Indies*), sendo possível também obter todo o tipo de conteúdo relacionado a jogos em 2D e jogos 3D, como por exemplo *sprites*, conteúdos sonoros ou até mesmo pacotes completos desenvolvidos pela sua comunidade, estes são gratuitos quanto pagos (SANTOS, 2022).

#### 4.3.2 Open Game Art

*OpengameArt* é um site que ajuda programadores e artistas a colaborar na criação de conteúdo para videogames, fornecendo uma plataforma onde é possível os próprios artistas compartilharem e arquivarem a sua arte com programadores que podem acessá-los facilmente e gratuitamente. O site é sustentado através de doações da comunidade (OPENGAMEART, 2023).

### 4.4. COMPONENTES PARA O DESENVOLVIMENTO

A seguir será apresentado sobre os componentes para desenvolvimento do jogo “Crônicas de Caim”.

#### 4.4.1 Mecânicas

Para a realização de um jogo é necessário avaliar o envolvimento do jogador com o jogo. A partir disto quando o jogo apresenta ao jogador imerso no ambiente do jogo, um desafio, este lhe possibilita, por meio da manipulação de mecânicas juntamente com a combinação de regras, a experimentação de novos elementos capazes de lhe proporcionar uma experiência gratificante e prazerosa (SANTOS, 2010).

O jogo “Crônicas de Caim” possuirá diversas mecânicas desenvolvidas para a primeira versão demo, durante a sua *gameplay* durante os diálogos com o NPC, o mesmo irá repassar os comandos de cada uma conforme o jogo avança. Será listado as mecânicas as quais o jogador poderá desfrutar durante o tempo que permanecer jogando.

- **Movimentação básica do personagem:** O jogador poderá se mover com o personagem na vertical ou na horizontal.
- **Correr:** Ao clicar em uma tecla em específica o personagem poderá se deslocar mais rápido podendo seguir com a tecla apertada tanto para horizontal quanto vertical.
- **Rolar/Esquivar:** O jogador poderá rolar, ao utilizar essa técnica irá ajudar a se esquivar de ataques inimigos.
- **Cortar Árvore:** O jogador poderá selecionar um machado sendo possível realizar o corte de árvores, ao final será possível coletar madeira.
- **Cavar:** O jogador poderá cavar com uma pá em pontos específicos pré-definidos no mapa, ao finalizar irá deixar um buraco no local.

- Coletar água: O jogador poderá coletar água em locais pré-definidos no mapa.
- Regar: Após a coleta de água é possível utilizar o regador para regar, após uma quantidade específica de água despejada irá aparecer um insumo coletável.
- Pescar: O jogador poderá pescar em locais específicos, terá uma taxa calculada a partir de uma porcentagem de sucesso podendo ou não coletar o peixe ao final da ação do personagem.
- Construir: O personagem poderá construir em locais pré-definidos no mapa caso tenha insumos suficientes coletados das árvores derrubadas.
- Combate: O personagem poderá selecionar uma espada, podendo causar dano a inimigos estes podendo estar parados no mapa, ao chegar em uma distância em específica o inimigo persegue o personagem e irá atacá-lo, tanto o personagem quanto o inimigo terão uma vida base que será reduzida conforme ataques recebidos.
- Conversa com NPC: Ao selecionar uma tecla de ação será possível interagir com um personagem não jogável, o mesmo poderá trazer dicas para ajudar durante o jogo.

#### **4.4.2 Narrativa**

A relação entre jogos digitais e narrativa está constantemente evoluindo. O potencial para criar histórias cativantes para jogos digitais não depende apenas de animações de alta definição ou de caras produções em vídeo, mas sim a concepção de momentos dramáticos. Uma história bem definida e alinhada já é o suficiente para trazer imersão e motivação para o jogador (NEPOMUCENO, 2012).

O jogo “Crônicas de Caim” em sua versão demo apresenta de início apenas interações simples de ajuda, entre jogador e NPC, trazendo assim uma visão do que se espera no futuro.

#### **4.4.3 Sonorização**

O som em jogos digitais são utilizados de várias maneiras, seja para comunicação de fala, músicas ou sons do ambiente. Através do som é possível adquirir informações sobre o ambiente em propriedades como características de lugares, presença de animais ou objetos na cena (CARAMEZ, 2019).

Em relação ao projeto desenvolvido será implementado sons de fundo e sons relacionados às ações executadas durante a *gameplay* tornando assim o jogo mais agradável e interessante para o público.

## 5. RESULTADOS E DISCUSSÕES

O desenvolvimento do jogo “Crônicas de Caim” alcançou o seu objetivo inicial, foi desenvolvido uma demo na *engine Unity* utilizando a linguagem de programação C#, trazendo todas as mecânicas propostas e funcionais, no qual o jogador poderá utilizar durante a sua *gameplay*.

Neste capítulo será apresentado as etapas de concepção do jogo, divididas em subcapítulos, será também comentado brevemente sobre o seu desenvolvimento, tendo em vista que é um assunto amplo serão abordados as principais características do jogo e suas mecânicas desenvolvidas, estas quais utilizam a metodologia *Scrum*.

### 5.1 HISTÓRIA DO JOGO

Em caim o jogador controlará o personagem “Tyler” (Figura 1), um jovem que decide migrar da sua cidade natal (um grande e próximo centro urbano) a procura de um pouco de sossego e paz. “Tyler” escolhe a cidade de Caim como seu destino, uma cidade fantasiosa e misteriosa e, ao longo do jogo, ele terá a oportunidade de aprender sobre a cultura local e experimentar a vida na área rural da região.

Figura 3 - Personagem Principal Tyler



Fonte: O Autor

O objetivo do jogo só é alcançado quando o personagem finalizar todas as tarefas definidas durante a saga no mundo de Caim.

### 5.2 CENAS DO JOGO

Para a realização e criação das cenas do jogo, foi escolhido o estilo artístico com artes em *pixel art*, sendo assim cenários, personagens e *interfaces* foram feitas com a resolução de 16 pixels, assim trazendo uma aparência “retrô” ao jogo desenvolvido. Foi utilizado artes com cores leves deixando o jogo, de certa forma com uma “aparência infantil” para atrair jogadores de todas as idades assim se adequando a diversas faixas etárias.

O jogo terá na versão demo 3 cenários, sendo o primeiro a entrada na cidade de Caim, o segundo se passará em uma floresta, ao final dela haverá uma área rural, a terceira parte apresentará uma pequena parte da cidade.



Figura 4 - Início do jogo



Fonte: O Autor

### 5.3 PERSONAGENS

Assim como as cenas do jogo, os personagens terão uma aparência estética infantil, trazendo rostos com um *design* simples e cores vibrantes. Para a demo desenvolvida será apresentado o personagem principal intitulado como “Tyler” e um NPC intitulado como “Wilson Braga”, este no qual irá ajudar o jogador com dicas no início do jogo, assim facilitando o aprendizado das mecânicas propostas. Ambos os personagens foram criados e idealizados pelo autor.

Haverá um inimigo que estará em pontos específicos no mapa, este com a aparência de um Goblin Verde, ele poderá atacar com uma espada o personagem principal. Este personagem foi utilizado através de sprints comprados no site ITCH.io.

Figura 5 - Personagens: Wilson e Inimigo Goblin



Fonte: O Autor

## 5.4 DESENVOLVIMENTO DAS SPRINTS

Para a realização das mecânicas do jogo foi utilizado a metodologia *Scrum*, primeiramente foi determinado todas as tarefas a serem desenvolvidas com prazos pré determinados. O *Scrum* é utilizado em projetos com planejamento complexo e geralmente utilizado por equipes, no jogo “Crônicas de Caim” foi adaptado e separado as mecânicas dividindo-as em 4 *sprints*, estas tarefas foram então realizadas entre os prazos estipulados variando entre 1 a 3 semanas.

As técnicas utilizadas para organizar o cronograma para o desenvolvimento foram:

- A cada *sprint* finalizada, a mesma era movida para um quadro chamado “Fase de Testes”.
- Após realizar os testes e verificar que a funcionalidade havia sido entregue e a mesma estava funcional era movida para um quadro nomeado “Concluído”
- Problemas durante a execução das *sprints* deveriam ser analisados e corrigidos o quanto antes para não atrasar as outras tarefas.

A seguir será explicado brevemente sobre cada *sprint* mostrando o seu processo de desenvolvimento com suas respectivas características.

### 5.4.1 Movimentação do personagem, Sistema de Diálogos e Telas do jogo e Cenário

O projeto começou com a criação do mesmo na *Unity*, na sequência foi realizado a criação de pastas para manter a organização do projeto conforme há o avanço do mesmo.

Após a criação das pastas foi realizado a importação dos *assets*, que são elementos que constituem um jogo para realizar a criação do cenário, personagem e demais itens que fazem parte do mesmo, estes foram importados para o *Unity* em suas respectivas pastas. Dentre os *assets* teve alguns que foram adquiridos pelo site *ITCH.io* utilizados e adaptados, e outros do site *Open Game Art* de forma gratuita.

Foi necessário realizar o *download* de dois itens encontrados dentro da loja da *Unity* no próprio editor, estes nomeados como *2D Sprite* responsável por realizar o ajuste e o recorte das imagens em *pixels*, estas das quais serão utilizadas no jogo e o *2D Tilemap Editor* responsável pela criação de *tilesets* este utilizado para transformar uma imagem em quadrados que farão parte da cena, ou seja após a junção por meio de uma imagem importada é possível criar um cenário.

No *Unity* foi criado um componente chamado *Grid*, este possui a função de criar camadas para inserir os *Tilesets*, estas camadas foram denominadas como chão, grama, paredes, água, enfeites. Após este processo foi desenvolvido os cenários aplicando componentes de física que o *Unity* disponibiliza para que quando o personagem colida contra os mesmos e ele não os ultrapasse.

A tela de menu do jogo foi criada a partir de uma nova cena, sendo na *Unity* ativos que contêm todo ou a parte de um jogo ou aplicativo. O menu possui inicialmente dois botões, Iniciar o Jogo e Sair do Jogo. Para a realização do menu foi utilizado um elemento do *Unity* denominado como *Canvas*, este sendo um objeto utilizado para guardar elementos de *UI* (Interface de Usuário). Após este processo foi criado um *script* que realiza a troca de cena, por esse *script* foram criados os métodos que serão usados em outras telas, sendo eles, iniciar o jogo, fim de jogo, reiniciar o jogo e sair do jogo.

Para a movimentação do personagem primeiramente foi necessário realizar o ajustes das suas animações, a animação é composta por uma sequência de imagens, no *Unity* há um componente chamado *Animation*, por ele é possível realizar a junção destas imagens e realizar a transformação e a impressão do movimento através delas. Foram criados 2 *scripts*, um responsável pelo controle de todas as animações e outro relacionado a todas as ações que o personagem poderá fazer, denominada de “*Player*”.

Na Figura 7 é representada duas condições para realizar a movimentação do personagem, no primeiro IF o personagem caminha para a direita e no segundo ele se movimenta para a esquerda, o método `transform.eulerAngles` é responsável por realizar a rotação de um objeto, logo colocando o objeto com uma rotação de 180 graus no eixo Y, é possível mudar de direção.

Figura 6 - Exemplo Movimentação do Personagem

```
if (player.direction.x > 0)
{
    transform.eulerAngles = new Vector2(0, 0);
}

if (player.direction.x < 0)
{
    transform.eulerAngles = new Vector2(0, 180);
}
```

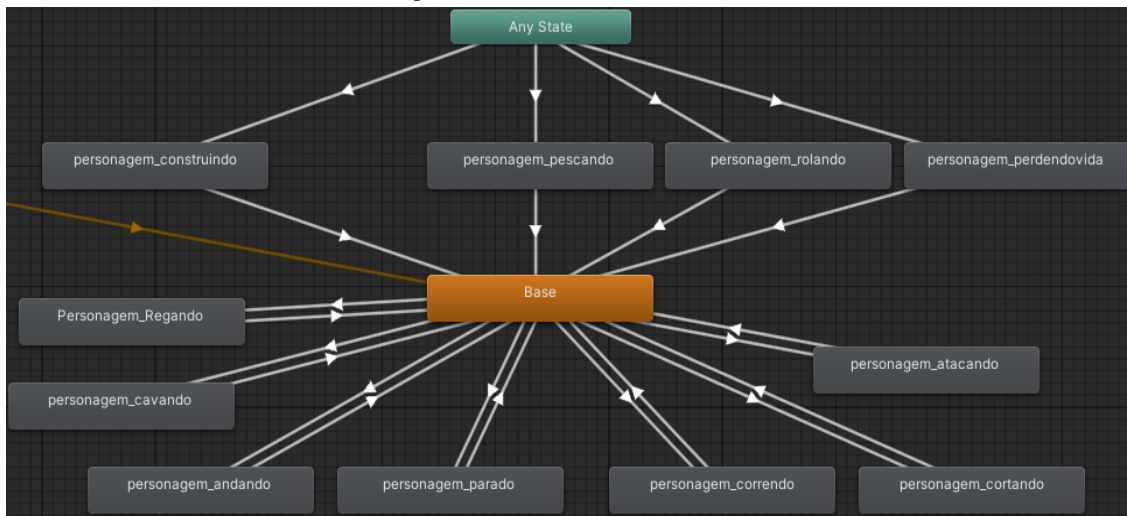
Fonte: O Autor

Na criação do NPC primeiramente foi necessário criar um *gameobject*, este objeto na *Unity* denominado na cena como “NPC\_1”, irá conter suas devidas animações e também um componente chamado *Box Collider*, este componente cria em volta do objeto um colisor. Após os ajustes foi inserido um *script* com o nome de “*Dialogo\_NPC*”, ele será o responsável por conter toda a lógica da fala entre o objeto “*Player*” e o “*NPC*”.

#### 5.4.2 Mecânica de Corte, Plantação, Dano ao Player, Ataque do Inimigo e Sons

Na segunda *sprint* inicialmente foram criadas as animações do personagem cortando, e na sequência as animações da árvore, foram criadas 3 animações para a árvore denominadas como, “*Arvore\_Parada*”, “*Arvore\_Mexendo*” e “*Arvore\_Cortada*”. Para realizar a troca de uma animação foi utilizada a aba *Animator*, que permite gerenciar quando cada animação deve ser executada. Por essa aba é possível ter acesso a todas as animações criadas, na sequência foi criado um parâmetro do tipo inteiro chamado transição no objeto “*Player*”, essa transição será a responsável por realizar a alteração das animações do personagem, a lógica das transições se encontram no *script* denominado “*AnimacoesdoPlayer*”, por ela é possível realizar a comparação quando o número for igual ou diferente da ação executada, todas as mecânicas executadas pelo jogador estarão nesta aba do *Animator*.

Figura 7 - Mecânicas no Animator



Fonte: O Autor

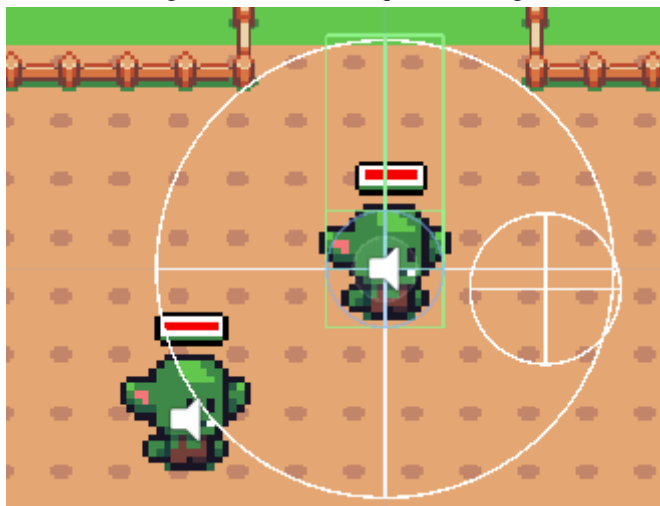
Após finalizado as animações junto com programação da árvore sendo cortada foi criado um objeto nomeado como “Madeira”, esse objeto tem intuito de aparecer na cena quando a árvore for cortada, então o jogador pode coletar para que depois possa realizar a mecânica de construção utilizando esse recurso obtido. Para esse objeto foi criado o *script* denominado como “Madeira”, a lógica contida faz com que sempre quando a animação da árvore cortada aparecer, será instanciado a madeira para o lado esquerdo da árvore, e quando o personagem passar por cima a madeira irá desaparecer e somar a mesma no objeto “Player”, dando a impressão de coleta do item e armazenando a quantidade no objeto.

A próxima mecânica desenvolvida foi a mecânica de plantação, primeiramente foi ajustado a animação do personagem do objeto “Player” na aba *animator*, foi criado um objeto chamado “colheita”, dentro deste objeto foi criado um objeto do tipo *2D Object* nomeado como “Slot”, nele foi colocado um *Box Collider*, este na *Unity* sendo um componente onde um objeto pode colidir e criado um *script* chamado “Slot\_Farm”, este no qual possui o objetivo de trazer uma imagem de um buraco ao realizar a animação com a pá. Após isso foi ajustado as animações do jogador jogando água com um regador, e no mesmo *script* foi realizado a programação para ativar a animação de coleta água quando identificar um colisor em um determinado local. Ao identificar a animação despejando a água em um buraco cavado, aparecerá um insumo no local, sendo possível coletá-lo e adicionando o mesmo no objeto “Player” através de um contador, similar ao método utilizado para a coleta de madeira.

Para o próximo desenvolvimento da *sprint*, foram ajustadas as animações do personagem atacando, e na sequência do inimigo atacando, dentro do objeto do inimigo foi criado um outro objeto chamado “PontoAtaque”, este objeto posicionado a frente do inimigo, e dentro deste objeto do inimigo foi criado 2 *Scripts*, um chamado como “Inimigo”, e outro chamado de “ControleAnimacaoInimigo”, tendo o objetivo receber as animações do mesmo. Foi criado 2 métodos no inimigo sendo o método atacar que ao realizar um ataque ativa a animação do objeto “Player” recebendo dano e assim perdendo vida, e o método “RecebeDano”, que ao receber o dano a barra de vida acima dele irá descendo, ao chegar no 0 aparecerá uma animação e o inimigo irá sumir da

cena. O inimigo terá um componente *BoxCollider* a sua frente e quando identificar que o objeto “*Player*” está à sua frente será executado uma animação de ataque, também foi ajustado para seguir o personagem quando estiver em uma certa distância assim trazendo uma *gameplay* mais divertida.

Figura 8 - Área de ataque do inimigo



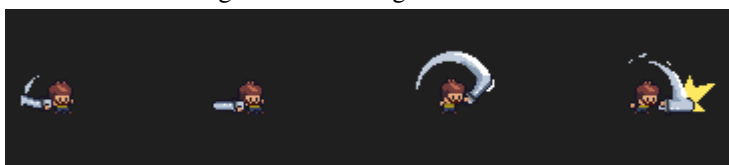
Fonte: O Autor

Por fim foi criado um objeto com o Nome de “Gerenciadordeaudio”, este possui possuíra um componente chamado *Audio Source* o qual tem o objetivo de adicionar sons para os objetos em cena deixando o jogo mais interessante. Uma das músicas que ficará tocando durante o jogo foi composta, demais sons foram baixados do site *opengameart*.

#### 5.4.3 Ataque do Player e Dano no Inimigo, Mecânica de Pesca e Coleta de Peixes

Para a terceira *sprint* foi iniciada ajustando as animações do objeto “*Player*” atacando e na sequência recebendo dano, foi utilizado uma lógica similar ao do inimigo, sendo que no *script* “*Player*” foi criado um método para atacar, e um outro para receber dano. Foi criado um colisor a frente do objeto “*Player*”, quando a espada for selecionada, a mesma somente será ativada ao executar a animação de ataque.

Figura 9 - Personagem Atacando



Fonte: O Autor

A próxima mecânica criada foi uma mecânica para pesca, primeiramente foi ajustado as animações do personagem pescando, e na sequência foi ajustado as animações na aba *Animator*, foi criado um objeto com o nome de “Lancar\_na\_cena”, este foi colocado em um local específico para realizar a pescaria, dentro dele foi colocado um *Box Collider*. Na sequência foi criado um *script* chamado “Pescar”, dentro deste *script* foi realizado a lógica da pescaria, ao clicar o botão de ação o personagem

irá executar a animação de pesca, dentro desse *script* terá um método chamado “Lancar”, este método irá ter uma variável inteira com uma função “*Random.Range*”, essa função tem o objetivo de trazer um número aleatório entre 1 e 100, foi realizado uma lógica para calcular em porcentagem sendo assim, o personagem poderá ou não realizar a pesca.

Após finalizado a mecânica de pesca foi realizado a lógica de coleta do peixe sendo similar ao da madeira já criado anteriormente, o personagem ao passar em cima do peixe pescado o mesmo irá sumir e irá contabilizar no objeto “*Player*”.

Figura 10 - Personagem Pescando



Fonte: O Autor

#### 5.4.4 Mecânica de Construção e Criação de Interfaces

Para a última *sprint* foi realizada a mecânica de construção. Para a realização da mesma primeiramente foi ajustado a animação do objeto “*Player*” utilizando um martelo, na sequência foi criado um objeto com o nome de “Casa”, neste objeto foi criado um componente do tipo *Sprite* e anexado a ele uma imagem da construção a ser executada, após isto foi adicionado na mesma um componente *Box Collider* para o objeto “*Player*” não conseguir passar pelo objeto “Casa”. Neste mesmo objeto foi criado um novo objeto com o nome de “Ponto\_inicial” e ajustado ao lado do local onde o mesmo será construído. Então foi criado um *Script* chamado de “Casa”, nele foi realizada a lógica para a construção. Para a animação ser executada o objeto “*Player*” precisará de uma quantidade específica de madeira coletada, somente após ele possuir essa quantidade é que será possível construir a mesma.

Figura 11 - Personagem Construindo



Fonte: O autor

Em jogos digitais a *interface* possui grande importância, ela preenche a lacuna afetiva entre a tecnologia e atividade humana, trazendo assim as necessidades do usuário de forma agradável (SEKHAVAT. SISI e ROOHI, 2021).

Como um dos últimos ajustes ao jogo desenvolvido foi criada uma *interface* para ajudar o jogador a identificar a ferramenta ou item de ataque que será utilizada durante o jogo na parte superior esquerda, também foi desenvolvida uma barra de vida referente ao personagem na parte superior direita e por fim uma barra contendo os itens coletáveis do jogo na parte inferior direita.

Figura 12 - Interface do Jogo



Fonte: O Autor

## 5.5 CONTROLES DO JOGO

A seguir, no Quadro 3, será apresentado sobre cada comando que o jogador poderá realizar durante o jogo e uma breve descrição explicando sobre a sua funcionalidade.

Quadro 3 - Comandos do jogo

Comando	Descrição
Botão direito do Mouse	Utilizado para realizar as ações conforme a ferramenta ou arma selecionada. Usado também para clicar nos botões nas telas do Menu Iniciar e Game Over.
Botão Esquerdo do Mouse	O personagem irá realizar um comando de esquiva/rolagem
Botão numérico 1 - Parte superior do teclado	Usado para selecionar a Ferramenta “Machado”
Botão numérico 2 - Parte superior do teclado	Usado para selecionar a Ferramenta “Pá”
Botão numérico 3 - Parte superior do teclado	Usado para selecionar a Ferramenta “Regador”
Botão numérico 4 - Parte superior do teclado	Usado para selecionar a arma “Espada”
Tecla E	Utilizada para realizar as ações do personagem, sendo: Interagir com o NPC, realização da coleta de água no regador, coleta do item plantado, utilização da vara de pescar e realização de uma construção.

Fonte: O autor

## 5.6 FLUXO DO JOGO (TELAS)

No jogo “Crônicas de Caim” até o momento foram desenvolvidas apenas três telas sendo a tela de Menu, a tela de *Pause* e a tela de finalização do jogo, abaixo será apresentado e explicado brevemente



### 5.6.1 Tela de Menu

Para a inicialização do jogo foi criada uma tela simples de menu, contendo 2 botões, sendo um para iniciar o jogo e outro para sair do jogo.

Figura 13 - Tela de Menu



Fonte: O Autor

### 5.6.2 Tela de Pause

Para a tela de *pause* foi criada uma tela simples, essa tela aparecerá sempre que o jogador clicar na tecla “P”, podendo utilizar a mesma tecla para voltar ao jogo. A tela terá um botão para voltar ao menu do jogo. O jogador não conseguirá executar nenhuma ação enquanto estiver nesta tela.

Figura 14 - Tela de Pause



Fonte: O Autor

### 5.6.3 Tela de Finalização do Jogo

Esta tela somente irá aparecer após o jogador chegar ao final da demo proposta, nesta tela terá um botão para voltar ao menu do jogo.

Figura 15 - Tela de fim de jogo



Fonte: O Autor

### 5.7 DISPONIBILIZAÇÃO DO JOGO

O Jogo “Crônicas de Caim” inicialmente foi disponibilizado através do repositório do *Google Drive* pelo *e-mail* do autor criado pela instituição. Futuramente o jogo será hospedado na plataforma *ITCH.io* para avaliações do público em geral. O jogo poderá ser baixado pelo Código (*QR CODE*), o mesmo estará zipado, bastando apenas descompactar em uma máquina com o sistema operacional *Windows* e então executar o arquivo “Caim Game”, na sequência o jogo irá iniciar.

Figura 16 - QR CODE



Fonte: O Autor

## 5.8 CONCLUSÃO DO CAPÍTULO

Por meio da utilização das técnicas utilizadas do método *Scrum* foi possível acompanhar o estado do jogo e então analisar se as tarefas estavam ou não sendo realizadas no prazo estipulado, assim podendo analisar o que foi feito até então. A ferramenta *Trello* ajudou na organização do projeto, auxiliando de maneira fácil e eficiente o gerenciamento do projeto, pois através dele foi possível visualizar o que estava sendo feito durante os prazos definidos. Sendo assim o projeto obteve êxito, e foi concluído o que foi definido no começo do projeto.

## 6. CONSIDERAÇÕES FINAIS

Por meio deste trabalho foi desenvolvido o jogo denominado como “Crônicas de Caim”, o qual possui o objetivo de trazer entretenimento para futuros jogadores e também despertar o interesse de acadêmicos para desenvolver projetos similares. O jogo já está com todas as mecânicas iniciais propostas implementadas e funcionais, trazendo animações suaves e uma *interface* dinâmica, sendo disponibilizado uma versão demo para testes, mostrando tudo o que o jogo irá oferecer futuramente.

É esperado que o desenvolvimento do projeto continue, será implementado novos cenários e também adicionado novas funcionalidades, sendo ajustado os comandos com *interfaces* visuais. As telas de menu, pause e finalização do jogo, serão redefinidas para ficarem mais atraentes ao público, também será adicionado possibilidades de combates com progressão de jogo contando com uma tela de *game over*, e também configurações adicionais podendo trocar o idioma do jogo, mudar opções de volume e sons do jogo, tendo o objetivo de despertar a curiosidade e interesse do público. O jogo possui planos de expansão para dispositivos móveis com o sistema operacional *Android*.

No futuro vai ser realizado o desenvolvimento de um *save game* para poder salvar e continuar onde o jogador parou e também a criação de novos inimigos incluindo chefes com uma dificuldade maior assim deixando a *gameplay* do jogo mais duradoura e divertida. O projeto desenvolvido até o momento fará parte de um portfólio pessoal.

## REFERÊNCIAS

- ACTON, M.; GAMES, I. Data-oriented design and c++. Luento. CppCon, 2014.
- ALENCAR, Filipe Henrique Bezerra Matos de. Pixel Art & Low Poly Art: catalisação criativa e a poética da nostalgia. 2017. 203 f. Dissertação (Mestrado) - Curso de Programa de Pós-graduação em Arte, Instituto de Artes - Ida, Universidade de Brasília, Brasília, 2017.
- ASEPRITE 2023. Disponível em: <<https://www.aseprite.org/>> Acesso em: 29/10/2023 .
- AZEVEDO, V. de A. (2012). Jogos eletrônicos e educação: construindo um roteiro para sua análise pedagógica. Dissertação (Mestrado em Educação). Programa de Pós-Graduação em Educação, Universidade Federal de Santa Catarina, Florianópolis. Disponível em: <http://repositorio.ufsc.br/xmlui/handle/123456789/96260>. Acesso em: 15 set. 2022.
- BANIQUED, P. L., Lee, H., Voss, M. W, Basak, C., Cosman, J. D., Souza, S., Severson, J., Salthouse, T. A., & Kramer, A. F. (2014). Selling points: what cognitive abilities are tapped by casual video games? *Acta Psychologica*, 142(1), 74-86. doi: 10.1016/j.actpsy.2012.11.009.
- BORGES<sup>1</sup>, Deise Miranda; BARREIRA<sup>1</sup>, Rafael Gonçalves; DE SOUZA<sup>1</sup>, Jackson Gomes. Comportamento de personagens em jogos de computador. 2009.
- CARAMEZ, Victor Freitas et al. Desenvolvimento de um protótipo de jogo 2D: Animação, Game Design e Sonorização. 2019.
- CAREGNATO, Sonia Elisa. Google Acadêmico como ferramenta para os estudos de citações: avaliação da precisão das buscas por autor. *Pontodeacesso*, v. 5, n. 3, p. 72-86, 2011.
- CHIES, Eduardo José. Perfect Weapon-Prototype: proporcionando competição e mantendo o engajamento dos jogadores.
- DANTAS, Marcos. Capitalismo na era das redes: trabalho, informação e valor no ciclo da comunicação produtiva. In: Lastres, H. M. M.; Albagli, S. (Org.). *Informação e globalização na era do conhecimento*. Rio de Janeiro: Campus, 1999. p. 216-261.
- DE CASTRO JUNQUEIRA, André Luis; SIRQUEIRA, Tassio Ferenzini Martins. Serious Games: Aplicação no ambiente educacional. *Caderno de Estudos em Engenharia de Software*, v. 4, n. 2, 2023.
- DOS SANTOS SOARES, Michel. Metodologias ágeis extreme programming e scrum para o desenvolvimento de software. *Revista Eletrônica de Sistemas de Informação*, v. 3, n. 1, 2004.
- GUIMARÃES, Bruna Barbosa et al. Dr. Solinho. 2022.

FEIJÓ, DIEGO, 2020, desenvolvimento de jogos em Java, TCC(graduação) - Universidade Federal de Santa Catarina. Centro Tecnológico. Sistemas de Informação. JTiled2D: uma biblioteca para desenvolvimento de jogos 2D em perspectiva top-down na linguagem Java.

FORBES, 2023, Promissor mercado de games.Disponívelem:<<https://forbes.com.br/>>. Acesso em: 10/08/2023.

GALVÃO, Giovane et al. HortiPrice: framework de domínio para formação de preço de venda da horticultura. 2020.

GULARTE, Daniel. Jogos eletrônicos: 50 anos de interação e diversão.Teresópolis: Novas Idéias, 2010. 192 p.

GRANJA, D. R. (2015). Indústria dos jogos eletrônicos: a evolução do valor da informação e a mais-valia 2.0.

GUERREIRO, Gustavo et al. Two Kingdoms: Relato de Desenvolvimento de um Jogo Utilizando Godot Engine. Anais do Computer on the Beach, v. 12, p. 571-574, 2021.

HARRIS, Blake J.. Console Wars: sega, nintendo, and the battle that defined a generation. New York: It Books, 2014.

HUNICKE, R.; LEBLANC, M. Z. Mda: A formal approach to game design and game research. In: . [S.l.: s.n.], 2004.

HUSSAIN, A.; SHAKEEL, H.; HUSSAIN, F.; UDDIN, N.; GHOURI, T. L. Unity Game Development Engine: A Technical Survey. p. 10, out. 2023.

ITAMARATY, 2022. Panorama Internacional de Jogos Eletrônicos. Disponível em: <[https://www.gov.br/mre/pt-br/assuntos/ciencia-tecnologia-e-inovacao/copy\\_of\\_PanoramaInternacionaldeMercadosdeJogosEletrnicos.pdf](https://www.gov.br/mre/pt-br/assuntos/ciencia-tecnologia-e-inovacao/copy_of_PanoramaInternacionaldeMercadosdeJogosEletrnicos.pdf)> Acesso em: 10/08/2023.

KOSTER, Raph Koster. A Theory of Fun for Game Design. Segunda Edição. O' Reilly Media, 2013.

LEITE, Patricia da Silva; MENDONÇA, Vinícius Godoy de. Diretrizes para game design de jogos educacionais. Proc. SBGames, Art Design Track, p. 132-141, 2013.

LEMES, David de Oliveira. Fundamentos metodológicos para criação, planejamento e desenvolvimento de jogos digitais. 2009. Tese de Doutorado. Dissertação de Mestrado, Pontifícia Universidade Católica de São Paulo, São Paulo.

LUCENA, Simone. Cultura digital, jogos eletrônicos e educação. 2014.

MICROSOFT, 2023. Visual Studio. Disponível em: <<https://visualstudio.microsoft.com/pt-br/vs/>>. Acesso em 15/10/2023.

NEPOMUCENO, Tainá Ribeiro. Narrativa e Formatação em roteiros de jogos eletrônicos. Orson–Revista dos Cursos de Cinema do Cearte, 2012.

OPENGAMEART 2023, Artes em projetos. Disponível em: <<https://opengameart.org/>> Acesso em: 29/10/2023 .

PANORAMA INTERNACIONAL DE MERCADOS DE JOGOS. Disponível em: <[https://www.gov.br/mre/pt-br/assuntos/ciencia-tecnologia-e-inovacao/copy\\_of\\_PanoramaInternacionaldeMercadosdeJogosEletronicos.pdf](https://www.gov.br/mre/pt-br/assuntos/ciencia-tecnologia-e-inovacao/copy_of_PanoramaInternacionaldeMercadosdeJogosEletronicos.pdf)>. Acesso em: 07/08/2023.

PASSOS, Erick Baptista et al. Tutorial: Desenvolvimento de jogos com unity 3d. In: VIII Brazilian Symposium on Games and Digital Entertainment. 2009. p. 1-30.

PERTERSON, K., Feldt, R., Mujtaba, S. and Mattsson, M. (2008) Systematic mapping studies in software engineering. In Proceedings of the international conference on Evaluation and Assessment in Software Engineering, 68-77.

PIXILART, 2023. An art community for everyone!. Disponível em: <<https://www.pixilart.com/>> Acesso em: 29/10/2023 .

SANTOS, Bruno Silva. Distribuição digital: explorando a ascensão dos gêneros de nicho por meio do Adventure Point-and-click.

SANTOS, Hélia Vannucchi de Almeida. A importância das regras e do gameplay no envolvimento do jogador de videogame. 2010. Tese de Doutorado. Universidade de São Paulo.

SANTOS, MATHEUS GOMES. INDÚSTRIA DE JOGOS DIGITAIS: paradigmas da indústria tradicional, desenvolvimento indie, e a plataforma itch. io. 2022.

SEKHAVAT, Y. A.; SISI, M. J.; ROOHI, S. Affective interaction: Using emotions as a user interface in games. Multimedia Tools and Applications, v. 80, n. 4, p. 5225–5253, 2021.

STEAM 2023, Gestão de direitos digitais. Disponível em: <<https://store.steampowered.com/about/>> Acesso em: 30/10/2023 .

STELLMAN, A.; GREENE, J. Learning Agile: Understanding Scrum, XP, Lean, and Kanban. Sebastopol: O'Reilly, 2015.

SUTHERLAND, J. Scrum: The Art of Doing Twice the Work in Half Time. Nova Iorque: Crown Business, 2014.

TRELLO 2023, Aplicativo para gerenciamento de projetos online . Disponível em: <<https://trello.com/home>> Acesso em: 02/11/2023 .

UNITY. Unity Manual. 2019. Disponível em:. Acesso em: 29/09/2023.

VISUAL STUDIO. docs. 2023. Disponível em: <<https://code.visualstudio.com/docs>>. Acesso em: 14 de agosto de 2023.