

TECNOLOGIA EM DESENVOLVIMENTO DE JOGOS DIGITAIS

HEITOR CASELLA, JUAN SOUZA, JOÃO ÁQUILA E MARIAH BACELAR

DREAMWALKERS

ARAÇATUBA – SP

2021

HEITOR CASELLA, JUAN SOUZA, JOÃO ÁQUILA E MARIAH BACELAR

DREAMWALKERS

Trabalho de conclusão de curso apresentado à
Universidade Católica Salesiano Auxilium,
como requisito para o recebimento do
Bacharel de Desenvolvimento de Jogos.

Orientador: Francis Martins de Souza

ARAÇATUBA – SP

2021

SUMÁRIO

1. RESUMO.....	1
2. ABSTRACT	1
3. INTRODUÇÃO	1
4. MATERIAIS E MÉTODOS	2
5. ENTENDENDO O SONHO	3
6. DESENVOLVIMENTO.....	4
6.1. Unity 2D	4
6.2. Tiled.....	5
6.3. Fade In	5
6.4. Aseprite.....	6
6.5. VSCode.....	6
7. CONCLUSÃO.....	7
8. BIBLIOGRAFIA	8

1. RESUMO

Apresentar uma história e deixar um jogador explorar o ambiente oferecido, desvendando mistérios e se conectando com o universo criado é algo que acompanha os jogos digitais desde o início. Jogos de exploração que contém *puzzles* como principal jogabilidade geralmente trazem consigo uma estória envolvente, buscando a imersão do jogador e também a criação de um vínculo entre jogador e personagens. Com isso em mente, o objetivo desse artigo é mostrar qual o processo de desenvolvimento envolvido nesse gênero de jogo. No caso de *Dreamwalkers*, um ambiente um pouco trágico junto com a carga emocional no relacionamento de duas irmãs serão a chave para um jogo tranquilo, porém forte, cheio de mistérios e quebra-cabeças.

Palavras chave: Desenvolvimento de Jogos; Puzzle; Aventura; Sonhos.

2. ABSTRACT

Presenting a story and letting a player explore the environment offered, unraveling mysteries and connecting with the created universe is something that has accompanied digital games since the beginning. Exploration games that have puzzles as the main gameplay usually bring with them an engaging story, seeking the immersion of the player and also the creation of a bond between player and characters. With that in mind, the purpose of this article is to show the development process in this game genre. In the case of *Dreamwalkers*, a somewhat tragic environment along with the emotional charge in the two sisters' relationship will be a key to a smooth but strong game, full of mysteries and puzzles.

Keywords: Game Development; Puzzle; Adventure; Dreams.

3. INTRODUÇÃO

“Em 50 anos de história, o mercado de jogos passou por uma transformação que o tirou do setor de brinquedos para produtos eletrônicos e mais recente, para o ambiente digital.” (The Shift, 2020).

A evolução do mercado de jogos é evidente, e boa parte da culpa dessa evolução é, além do avanço tecnológico, são os *puzzles* (quebra-cabeças, desafios) que são encontrados em cada gênero de jogo existente.

Apesar de existirem algumas exceções, boa parte dos jogos hoje em dia contam com uma exploração, seja em mundo aberto ou não, e um quebra cabeça para prender a atenção do jogador e render tempo de jogos.

Podemos ver a existência desse estilo de *gameplay* no jogo *Zelda Breath of The Wild* (2017) da gigante Nintendo. O jogo se passa num mundo totalmente aberto, você tem seus objetivos e seus momentos de contemplação da história, porém uma parte que se destaca no jogo são os santuários que além de oferecerem um combate, é necessário que você resolva uma série de quebra-cabeças usando as ferramentas que o jogo te proporciona para atingir o final e ganhar o tão esperado item de recompensa.

Outro bom exemplo é a franquia *God of War* (2005 – atualmente). Todos os jogos da franquia são totalmente focados em combate, popularmente conhecido como o gênero *hack'n'slash*, mas em muitos momentos o jogo simplesmente para e te joga no meio de um quebra-cabeça. Você se encontra na necessidade de interagir com o ambiente, achar alavancas, abrir portas, recuar um pouco no desenvolvimento do mapa, tudo isso para poder voltar à rotina de massacrar casualmente seus inimigos.

Tendo tais pontos em mente e estabelecido que puzzles são importantes pra maior parte de jogos, é necessário entender que com *Dreamwalkers* não seria diferente.

O projeto conta com os puzzles como mecânica principal, apesar de ter alguns lapsos de combate em determinados pontos, tudo isso enquanto envolve o jogador numa história melancólica entre duas irmãs com um fundo perturbado e que cabe ao jogador descobrir junto com elas qual é sua origem.

4. MATERIAIS E MÉTODOS

Para dar funcionalidade ao projeto, utilizaremos várias plataformas e aplicações que surgiram com o tempo com o objetivo de facilitar a vida daqueles que criam algo.

Para esboços, *concept arts* e arte do jogo em geral utilizaremos o Photoshop CC (Adobe) e o Aseprite (David Capello). O Photoshop será utilizado para

alguns sketches de personagem e cenário. O Aseprite por sua vez, será para transformar os esboços em Pixel Art.

Para organizar o projeto e criar todo o corpo do utilizaremos o motor de jogo Unity. Ele será responsável por juntar tudo que desenvolvemos por fora e dar vida ao projeto. Desde a forma que a cena é formada para o jogador até a interação da interface e algumas animações, tudo foi graças às ótimas funcionalidades dessa *game engine*.

Quanto à parte sonora, estaremos utilizando o FL Studio (Image-Line), por ser uma DAW (*Digital Audio Workstation*) com muitos tutoriais e com uma comunidade bem ativa, além de ter uma breve curva de aprendizado.

5. ENTENDENDO O SONHO

O sonho tem sido um mistério para os humanos desde o início dos tempos. Para algumas culturas todo sonho indica algo que pode acontecer ou reflete algo que aconteceu na sua vida. Para outras, o sonho pode ser visto como um presságio ou um “feedback” de algo que está acontecendo, porém você ainda não notou.

Para a psicanálise, o sonho começou a ser um instrumento de estudo com Sigmund Freud (1856-1939) em 1900, durante a publicação de seu livro “A interpretação dos Sonhos”.

“O sonho para a psicanálise tem como um dos principais intuitos destacar a importância que o que sonhamos pode ter em nossas vidas.” (Psicanálise Clínica, 2017).

Segundo o que foi proposto por Freud em seu livro, o sonho era o local onde os nossos desejos mais profundos se manifestavam, sejam desejos que temos conhecimento ou até mesmo aqueles que não temos noção, e nesse local eles se realizavam.

Outra grande mente que estudava o significado dos sonhos era Carl Jung (1875-1961). Ele por sua vez se distanciou um pouco da ideia de Freud, pois não concordava que a psique apenas se resumiria a um disfarce de um desejo reprimido.

“Carl Jung não reduz os sonhos à satisfação de desejos reprimidos no inconsciente pessoal, como o fez Freud. Ele os toma como mensageiros de complexos.” (Regina Maria Faria Gomes).

Segundo Jung, temos um segundo “alguém” dentro de nossa consciência, e sua forma de se comunicar conosco são os sonhos. Essa consciência por sua vez seria igual para todos os indivíduos.

Dreamwalkers tenta misturar as duas teorias, trazendo tanto a ideia de Freud, de desejos reprimidos (problemas que os sonhadores enfrentam), e também a ideia de Jung (as personagens principais, que andam pelos sonhos ajudando os sonhadores).

6. DESENVOLVIMENTO

Para desenvolvermos o jogo, tivemos de tomar decisões de quais Softwares iríamos utilizar, escolhemos os que mais nos daria produtividade para desenvolvermos algo bom em quesitos de história, ambientação e manuseio do mesmo. A seguir será apresentado quais ferramentas foram utilizadas para desenvolvermos o jogo.

6.1. Unity 2D

Como IDE para desenvolvermos o jogo, utilizaremos a Unity, por ser uma IDE leve, simples e muito explorada pela comunidade, utilizando a linguagem de programação C#, conseguimos fazer simples ações com apenas alguns comandos e deixar que a Unity faça o resto. Com ela é possível criar jogos tanto 3D como 2D, nossa opção foi de fazer um meio termo intitulado de 2.5D, chamado também de visão isométrica, poucos desenvolvem diretamente para esse estilo de jogo, a própria comunidade da Unity não explora muito o lado do 2.5D, por esse fato tivemos de explorar outros softwares que nos ajudasse a criar uma ambientação para o jogo em si, porém é uma ótima visão para RPGs em pixel.

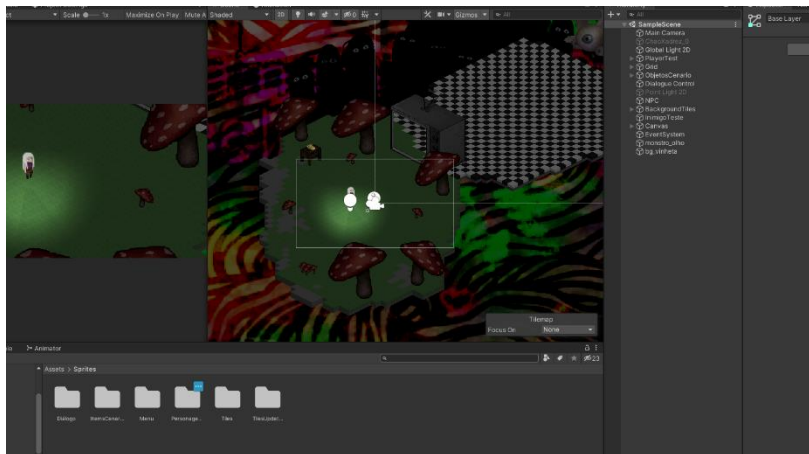


Figura 1 – Visão da página inicial do projeto na Unity.

6.2. Tiled

Seguindo com a ideia de um jogo 2.5D, o Tiled é um editor de mapa criado para games do tipo RPG, versátil e que possui várias funcionalidades, para a criação e uma visão geral de como nosso mapa poderia ficar, utilizaremos o mesmo por possuir a concepção de visão isométrica, além de podermos importar várias imagens personalizadas e exportar na resolução que desejarmos.

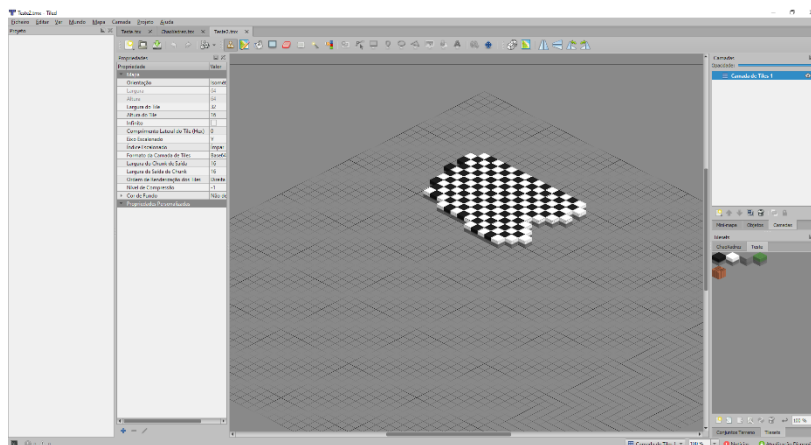


Figura 2 – Tiled com testes da ambientação do jogo.

6.3. Fade In

Direcionado para autores-roteiristas que desejam escrever seus roteiros para projetos, cinema e até mesmo dentro do jogo, esse software utiliza muitas formatações para especificar e detalhar o máximo possível do decorrer da história, ambientação que os personagens estão, características físicas, pensamentos e várias outras declarações podem ser dadas para uma cena, para um jogo, é mais utilizado para montagem de uma cutscene ou para servir de guia de diálogos entre os personagens e até mesmo onde os mesmos se encontram.

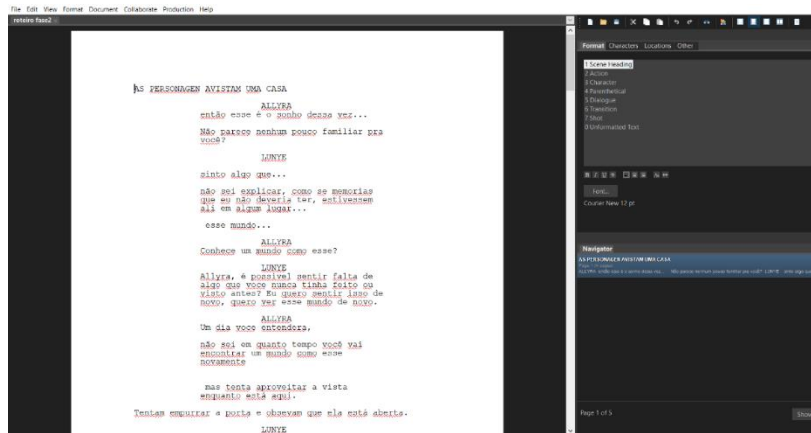


Figura 3 – Roteiro inicial da história.

6.4. Aseprite

O Aseprite, é uma ferramenta de edição de imagens de código aberto, projetado principalmente para desenhos e animações em pixel art, o mesmo possui várias ferramentas para edições em camadas, frames e titlemap. Com a escolha de fazer um jogo em pixel art, uma ferramenta completamente direcionada para isso foi de grande ajuda para o desenvolvimento do projeto em quesitos de produtividade.

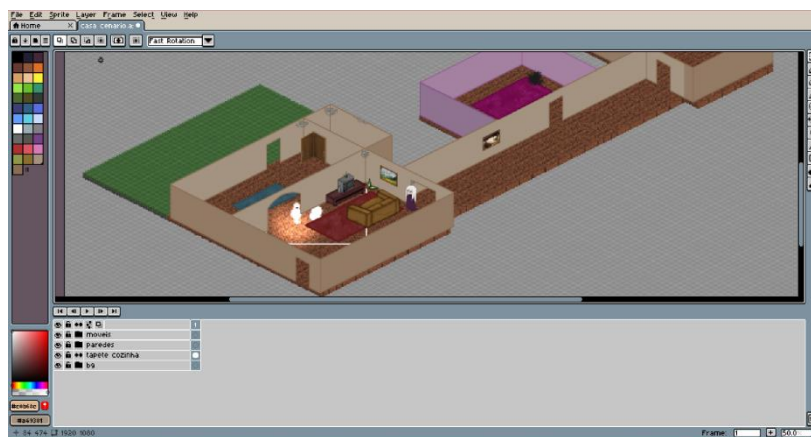


Figura 4 – Aseprite com elementos do cenário

6.5. VSCode

Além de todos esses elementos mais as ferramentas disponibilizadas pela própria engine, utilizaremos também o VSCode como principal meio de elaboração dos códigos. Apesar de ser considerado um editor de texto, essa ferramenta é muito mais leve se comparado com a IDE padrão da Unity, além de oferecer inúmeros plugins que auxiliam no desenvolvimento e criação de códigos dentro da Unity.

```

1 public void SetDialogueText(string text)
2 {
3     m_dialogueText = text;
4 }
5
6 public void PlayVoiceActor(AudioClip clip)
7 {
8     [SerializeField] AudioSource voiceActor;
9     voiceActor.Play();
10 }
11
12 public void PlayVoiceActor(AudioClip clip, string actorName)
13 {
14     [SerializeField] AudioSource voiceActor;
15     voiceActor.Play();
16 }
17
18 public void PlayVoiceActor(AudioClip clip, string actorName,
19                             string voiceActorName)
20 {
21     [SerializeField] AudioSource voiceActor;
22     voiceActor.Play();
23 }
24
25 public void PlayVoiceActor(AudioClip clip, string actorName,
26                             string voiceActorName,
27                             string voiceActorVolume)
28 {
29     [SerializeField] AudioSource voiceActor;
30     voiceActor.Play();
31 }
32
33 public void PlayVoiceActor(AudioClip clip, string actorName,
34                             string voiceActorName,
35                             string voiceActorVolume,
36                             string voiceActorPitch)
37 {
38     [SerializeField] AudioSource voiceActor;
39     voiceActor.Play();
40 }
41
42 public void PlayVoiceActor(AudioClip clip, string actorName,
43                             string voiceActorName,
44                             string voiceActorVolume,
45                             string voiceActorPitch,
46                             string voiceActorSpeed)
47 {
48     [SerializeField] AudioSource voiceActor;
49     voiceActor.Play();
50 }
51
52 public void PlayVoiceActor(AudioClip clip, string actorName,
53                             string voiceActorName,
54                             string voiceActorVolume,
55                             string voiceActorPitch,
56                             string voiceActorSpeed,
57                             string voiceActorVolumeMultiplier)
58 {
59     [SerializeField] AudioSource voiceActor;
60     voiceActor.Play();
61 }
62
63 public void PlayVoiceActor(AudioClip clip, string actorName,
64                             string voiceActorName,
65                             string voiceActorVolume,
66                             string voiceActorPitch,
67                             string voiceActorSpeed,
68                             string voiceActorVolumeMultiplier,
69                             string voiceActorPitchMultiplier)
70 {
71     [SerializeField] AudioSource voiceActor;
72     voiceActor.Play();
73 }
74
75 public void PlayVoiceActor(AudioClip clip, string actorName,
76                             string voiceActorName,
77                             string voiceActorVolume,
78                             string voiceActorPitch,
79                             string voiceActorSpeed,
80                             string voiceActorVolumeMultiplier,
81                             string voiceActorPitchMultiplier,
82                             string voiceActorSpeedMultiplier)
83 {
84     [SerializeField] AudioSource voiceActor;
85     voiceActor.Play();
86 }
87
88 public void PlayVoiceActor(AudioClip clip, string actorName,
89                             string voiceActorName,
90                             string voiceActorVolume,
91                             string voiceActorPitch,
92                             string voiceActorSpeed,
93                             string voiceActorVolumeMultiplier,
94                             string voiceActorPitchMultiplier,
95                             string voiceActorSpeedMultiplier,
96                             string voiceActorVolumeMultiplierMultiplier)
97 {
98     [SerializeField] AudioSource voiceActor;
99     voiceActor.Play();
100 }

```

Figura 5 – Código que trata o diálogo dos personagens

7. CONCLUSÃO

Conforme o exposto no decorrer deste artigo, onde pontuamos os passos executados para o desenvolvimento de nosso projeto, pudemos concluir que, jogos que utilizam da visão 2.5D tem uma gama um pouco menor de conteúdo na internet, a dificuldade de achar conteúdo para ser abstraído não é tão fácil quanto parece.

O tema escolhido para a história também demonstrou ser de grande relevância para explorarmos um pouco mais da psicologia dos sonhos, psicologia essa que possui muitas perguntas até os dias atuais, devido sua enorme carga de possibilidades, além de trazer ao jogador várias experiências dependendo do tipo de situação em que o sonho em questão disponibilizava para as personagens.

8. BIBLIOGRAFIA

REDAÇÃO, T.S. The Shift Info. O boom do mercado de games, 2020. Disponível em: <<https://theshift.info/hot/o-boom-do-mercado-de-games>>. Acesso em: 23 Abril 2021.

CLÍNICA, P. Psicanálise. Psicanálise Clínica, 2017 < <https://www.psicanaliseclinica.com/o-que-e-sonho-para-psicanalise>>. Acesso em 31 Setembro 2021.

CHENIAUX, E. Scielo Brasil. Os sonhos: integrando as visões psicanalítica e neurocientífica, 2006. Disponível em < <https://www.scielo.br/j/rprs/a/BqyRYJPBCVrCwmSnY5JcMHj/?lang=pt>>. Acesso em 28 Julho 2021.

PSICOLOGIA. A mente Maravilhosa. A simbologia dos sonhos na análise junguiana, 2019. Disponível em < <https://amenteemaravilhosa.com.br/simbologia-dos-sonhos-junguiana/>>. Acesso em 28 Julho 2021.

SOUZA, F. Psicologia MSN. Os significados dos sonhos para C. G. Jung, 2014. Disponível em < <https://www.psicologiamsn.com/2014/01/os-significados-dos-sonhos-para-c-g-jung.html>>. Acesso em 16 Março 2021.

SOFTWARE UNITY. Disponível em < <https://unity.com/pt>>. Acesso em 30 Abril 2021.

SOFTWARE PHOTOSHOP. Disponível em < https://www.adobe.com/br/products/photoshop/landpa.html?gclid=Cj0KCQiAy4eNBhCaARIsAFDVtI2xjJd-UMpf317ZKE0xSOzSk0hycJQmdLomAxORv1ky-JA-BCPineYaAvGrEALw_wcB&sdid=KQPOM&mv=search&ef_id=Cj0KCQiAy4eNBhCaARIsAFDVtI2xjJd-UMpf317ZKE0xSOzSk0hycJQmdLomAxORv1ky-JA-BCPineYaAvGrEALw_wcB:G:s&s_kwid=AL!3085!3!534509111683!e!!g!!fotoshop%20online!188192502!46790623599>. Acesso em 30 Abril 2021.

SOFTWARE ASEPRITE. Disponível em < <https://www.aseprite.org/>>. Acesso em 30 Abril 2021.

SOFTWARE VSCODE. Disponível em < <https://code.visualstudio.com/>>. Acesso em 30 Abril 2021.

SOFTWARE FADE IN. Disponível em < <https://www.fadeinpro.com/index.pl#who>>. Acesso em 30 Abril 2021.

SOFTWARE TILED. Disponível em < <https://www.mapeditor.org/>>. Acesso em 30 Abril 2021.