

**UNIVERSIDADE FEDERAL DE MINAS GERAIS
CINEMA DE ANIMAÇÃO E ARTES DIGITAIS**

Trabalho de conclusão de curso

**ANÁLISE DE PROBLEMAS ENFRENTADOS NA CRIAÇÃO DE UM JOGO 2D E
SOLUÇÕES QUE PODEM SER APLICADAS POR OUTROS DESIGNERS DE
JOGOS.**

Pedro Henrique Câmara Botelho

Orientador: Profa. Dra. Mariana Ribeiro da Silva Tavares

Belo Horizonte, 2022

PEDRO HENRIQUE CÂMARA BOTELHO

**ANÁLISE DE PROBLEMAS ENFRENTADOS NA CRIAÇÃO DE UM JOGO 2D E
SOLUÇÕES QUE PODEM SER APLICADAS POR OUTROS DESIGNERS DE
JOGOS.**

Artigo apresentado ao Curso de Cinema de Animação e Artes Digitais da Universidade Federal de Minas Gerais, como requisito parcial para obtenção do grau de Bacharel em Cinema de Animação e Artes Digitais da Escola de Belas Artes - UFMG.

Orientação: Profa.Dra. Mariana Ribeiro da Silva Tavares

Belo Horizonte, 2022

RESUMO

Esse trabalho analisa problemas enfrentados no processo de criação da *demo* do jogo independente em 2D, usando *pixel art*, *A scientifically Tested Game*. Desenvolvido por um grupo de três estudantes de graduação da UFMG: Matheus Leonel de Faria (Graduando em Cinema de animação e Artes Digitais), Vinicius Reis Camilo (Física licenciatura) e este autor. A *engine*¹ do jogo foi programada em linguagem D. O objetivo é compartilhar as soluções encontradas tornando mais claro o processo de criação de um jogo e evidenciando diferentes possibilidades e perspectivas desse processo, que podem ser úteis para futuros designers e programadores de jogos.

Palavras-chave: Video-game; narrativa interativa; pixel art, engine, roteiro, tile-set, design de personagem.

ABSTRACT

This work analyzes the problems faced in the process of creation of an independent game demo in 2D, using pixel art, *A scientifically Tested Game*. Developed by a group of three students of UFMG: Matheus Leonel de Faria (Graduating in animation cinema and digital arts), Vinicius Reis Camilo (Graduated in Physics) and this author. The engine was programmed in D language. The purpose is to share the solutions founded, making the process of creating a game more clear, showing different possibilities and perspectives of that process that can be useful for future game designers and programmers.

Keywords: Video-game; interactive narrative; pixel art, engine, script, tile-set, character design.

¹ *Engine*: Também conhecido como Motor de Jogo, é um programa desenvolvido para ser a estrutura do jogo, é a partir dele que o jogo é criado e estruturado.

SUMÁRIO

1. INTRODUÇÃO	5
2. METODOLOGIA	7
3. ENGINE	8
3.1 Uma breve história dos tiles	8
3.2 Problemas enfrentados no uso dos tiles	10
3.3 Alternativa para o uso de tiles	13
4. ROTEIRO	15
4.1 As convenções do roteiro para live actions	15
4.2 Soluções para como descrever no roteiro a imprevisibilidade dos video-games	16
5. DESIGN DE PERSONAGENS	18
5.1 Egon	19
5.2 Dice	21
5.3 Uder	21
5.4 Kenzlyn e Victor	22
6. CONSIDERAÇÕES FINAIS	23
7. REFERÊNCIAS	25

1. INTRODUÇÃO

O jogo *A scientifically tested game* é uma demo, com perspectiva *Top-Down*², do gênero *RPG* de ficção-científica, simulação e aventura. Ele foi produzido ao longo de três anos em paralelo ao Curso de Cinema de Animação e Artes Digitais e apresenta como personagens, um grupo de estudantes de física: Egon, Kenzlyn, Uder, Victor e Dice a partir da seguinte premissa: No decorrer de um experimento proposto pelo protagonista Egon, os personagens passam a compreender a física do universo do próprio jogo, se tornando capazes de controlá-lo. Esse poder impulsiona o surgimento das mecânicas e habilidades de cada personagem.

Esses estudantes estão prestes a se formar e Egon, sendo um deles está em busca de apoio para sua pesquisa e de um trabalho, ele fracassa nos dois, já que durante seu tempo na universidade ele desenvolveu inimizade com a maior parte das autoridades, como os professores, por conta de suas ideias e personalidade rebelde. Sendo assim, nenhum membro do corpo docente escreveu uma carta de recomendação e Egon sai toda manhã em busca de emprego. O jogo inicia exatamente em uma dessas manhãs, em que dessa vez, ao voltar frustrado para casa, Egon tem uma conversa com Uder, seu melhor amigo, que traz uma luz para a sua pesquisa, o elemento que faltava. Sua pesquisa é direcionado para o funcionamento do próprio tempo e o Egon tem a ideia de que o tempo de seu universo funciona assim como a mecânica quântica, em pacotes, dividida em números inteiros, ou seja, Egon está perto de descobrir que o universo do jogo funciona através de *frame rate*³, que o tempo de seu universo está em função do tempo da vida real, que o mundo dele é atualizado através de frames, esse é o primeiro passo para ele descobrir que ele está dentro de um jogo.

Egon então precisa provar isso através de um experimento que ele conduzirá através de uma máquina, porém, para que isso ocorra, ele precisa de outros especialistas da física, como eletromagnetismo, mecânica quântica, etc, que são as especialidades dos outros personagens. É nesse momento que Egon precisa deixar seu orgulho de lado e pedir ajuda aos outros estudantes, deixar o passado, suas brigas e conflitos em prol de seu objetivo, de seu sonho. No decorrer da *demo* o jogador irá atrás desses outros personagens e reuni-los para o grande experimento.

² *Top-down*: Jogos em que a perspectiva da visão do jogador se dá acima do que ocorre no jogo, olhando para baixo. Exemplos: *Grand Theft Auto* (1997), *Final Fantasy VI* (1994), *Hyper Light Drifter* (2016), *Pokémon Ruby e Sapphire* (2002).

³ *Frame-rate*: É o número de atualizações que o monitor faz em função do tempo, por exemplo: 24 frames por segundo. Significa que o monitor atualizou a imagem exposta 24 vezes no tempo de 1 segundo.

A estrutura do jogo foi pensada a partir desta premissa e de como ela seria representada em termos visuais e de *gameplay*. A partir dessa proposta surgiu a ideia de *gameplay*; dos poderes de cada personagem; de como funcionaria o sistema de RPG, o que corrobora o pensamento do designer de jogos e escritor Lee Sheldon:

Os desenvolvedores ficam lá sentados em suas salas de conferência por semanas, cada dia com uma ideia “nunca antes feita” de *gameplay*, uma atrás da outra, vendo os protótipos sendo criados na tela de plasma fixada na parede, e então de cada um deles percebe: “Ei! Nós não dedicamos muito tempo na construção da história . . . (SHELDON, 2004).⁴

Não é o ideal ter apenas um conceito de jogabilidade⁵ por si só, é necessário ter um contexto para as mecânicas do jogo, nem que este contexto seja superficial. Por isso em *A scientifically tested game*, a forma de se jogar o jogo foi melhor visualizada depois que uma estrutura mínima para história foi estabelecida, e os personagens definidos - cada um com suas peculiaridades e habilidades relacionadas aos campos da física. Isso possibilitou pensar na atribuição de poderes aos personagens como o do protagonista Egon, com a capacidade de parar o tempo para não ser atingido por projéteis ou algo que possa dar dano ao personagem. Ou ainda, o poder do personagem Kenzlyn de manipular a eletricidade e resolver *puzzles* no jogo que envolvem fios e cargas elétricas, como bobinas de Tesla. Cada aspecto do jogo foi desenvolvido de forma simultânea, com todos os envolvidos no processo, cientes de todo o trabalho. O que reforça a ideia da Diretora de arte da empresa Insomniac Games, Jacinta Chew no *art-book* de *Marvel's Spider-man* (2018):

Ideias que são bem pensadas em conjunto de toda a equipe, são aquelas que tendem a sobreviver por mais tempo no decorrer da criação do jogo. Assim que eu tenho uma boa ideia de *gameplay* e história que envolve o personagem, ambiente, objetos, eu dou aos *concept artists* uma breve descrição. (CHEW, 2018)

Na criação do jogo, foram estipulados prazos e metas com o objetivo de produzir uma demo com o prólogo do jogo, de forma a demonstrar ao público seu conteúdo e parte das

⁴ The developers have been sitting around in conference rooms for weeks coming up with one great original never-done-before idea for *gameplay* after another, watching each build of the prototype on the plasma screen on the wall, and all of a sudden one of them realizes, “Hey, we haven’t spent a lot of time on the story . . . (tradução nossa)

⁵ Jogabilidade: São todas as experiências do jogador ao interagir com as mecânicas propostas pelo jogo, a forma de jogá-lo.

mecânicas de combate RPG presentes. Após a conclusão da demo, o objetivo foi divulgar o projeto no intuito de conseguir apoio externo para seu desenvolvimento.

A scientifically tested game foi desenvolvido por um grupo de três estudantes de graduação da UFMG: Matheus Leonel de Faria (Graduando em Cinema de animação e Artes Digitais), Vinicius Reis Camilo (Física licenciatura) e este autor, encarregado por escrever o roteiro; criar o conceito visual; as animações; organizar o projeto e produzi-lo. Matheus Leonel ficou responsável pela criação e gravação da parte sonora - sons de diálogo dos personagens, composição das músicas e sons ambiente. Vinicius Camilo, responsável pela programação da engine em linguagem D e incorporação de todos os elementos no jogo: cenários, animações, sons, etc.

Os problemas enfrentados e a forma com que o grupo lidou com eles, moldaram o resultado final. Os problemas que serão elencados neste artigo foram selecionados tendo em vista os aspectos mais relevantes para a construção do projeto. A análise dos problemas e soluções foram organizados nas categorias: *engine*, roteiro e design de personagens. Após escolhermos a *engine* que foi programada em linguagem D, nós nos deparamos com outras questões, como o uso de *tile-maps*, estruturação de roteiro para o jogo e questões visuais como o design de personagens.

2. METODOLOGIA

Esse artigo analisa os problemas enfrentados na criação de um jogo 2D, indie, *top-down* em *pixel art* elencando possíveis soluções e formas de lidar com esses obstáculos no decorrer do processo de produção. O objetivo é tornar mais claro o processo de criação para aqueles que almejam seguir o mesmo caminho na criação de jogos e proporcionar outras perspectivas.

Os problemas abordados se relacionam com o processo artístico e foram organizados nas categorias:

- a) Engine, abordando o uso de *tile-set* e *tile-map* em diálogo com o autor Brett Camper, engenheiro de softwares .
- b) Roteiro, em diálogo com o autor Syd Field evidenciando as limitações dos paradigmas apontados por Field quando aplicados a video-games.
- c) Design dos personagens, apresentando os desafios na criação de suas personalidades através dos retratos de diálogo - uma vez que a resolução das imagens no decorrer do gameplay não permite expressar com qualidade parte de suas feições. Para a análise dos desafios na criação desses designs nos amparamos em algumas idéias da autora

Katherine Isbister, pesquisadora no laboratório de pesquisa social e comportamental e professora no departamento de linguagem, literatura e comunicações no instituto politécnico Rensselaer.

3. ENGINE

3.1 *Uma breve história dos tiles*

No projeto inicial para a elaboração do jogo pretendia-se utilizar dois *softwares*: Tiled⁶ e Unity⁷. A Unity seria a *engine* do jogo e o Tiled seria utilizado para a criação dos mapas, no intuito de facilitar a manipulação dos *tiles*.

O motivo dessa escolha foi a familiaridade desse autor com esses programas, mesmo não sendo ele o responsável final pela programação. Optou-se, então, pela programação em linguagem D, mais familiar ao programador Vinicius Camilo. A linguagem D é uma linguagem baseada em C++ e C, ela é prática, objetiva e melhor para compilar o programa de forma otimizada, tornando ela eficiente para a construção da engine. Apesar de serem descartados o uso da *Unity* e do *Tiled*, foi mantida a ideia do uso de *tile-maps* e *tile-sets* que são o recurso padrão para a criação de jogos de diversos gêneros, como: RPGs, Simulações (como por exemplo o jogo *Stardew Valley* de 2016), Isométricos (como por exemplo o jogo *Tactics Ogre: Let Us Cling Together* de 1995), etc. Seu surgimento se deu a partir de uma limitação tecnológica, antes mesmo dos anos de 1980, como lembra Brett Camper:

Tenha em mente que um jogo típico de Arcade tem aproximadamente 256 pixels por 256 pixels, então apenas para poder limpar o monitor requer $256 \times 256 = 65,536$ operações. Se cada operação levar 4 ciclos de clock, então isso é mais ou menos 1/4 de segundo em 1 MHz CPU! Isso é simplesmente demais para o processador acompanhar. (BRETT CAMPER apud AARON GILES, 2012, p. 173)⁸

Ou seja, 16,384 segundos, que são aproximadamente 273 minutos. Por isso os desenvolvedores criaram uma outra forma de dispor os pixels na tela, o modelo do *Tile-map*, um agrupamento de pixels que formam um quadrado, por exemplo 8x8, com o objetivo de otimizar e simplificar diversas funcionalidades tanto gráficas quanto mecânicas dos jogos. Um dos primeiros a fazer o uso desse recurso foi *Galaxian* de 1979 da Namco. O *Tile-map*,

⁶ *Tiled*: Um programa editor de levels que trabalha em conjunto de uma engine, para jogos que se fundamentam no uso de tiles.

⁷ *Unity*: *Engine* criada pela empresa *Unity Technologies*.

⁸ “Keep in mind that a typical arcade game is approximately 256 pixels by 256 pixels, so even just clearing the screen requires $256 \times 256 = 65,536$ operations. If each operation takes 4 clock cycles, then that is about 1/4 of a second on a 1 MHz CPU!” There are simply too many pixels for the processor to keep up. (tradução nossa)

portanto, restringe o espaço jogável e a liberdade do artista ao compor uma cena, o que pode ser vantajoso em alguns casos como em *Pokémon - Ruby version* (2002), um RPG, em que ao invés do computador contar cada passo do personagem dentro dos arbustos para iniciar um combate, ele contava os *tiles*, ou o número de arbustos.



Figura 1: Arbustos em Pokemon Ruby. Print Screen do jogo *Pokémon - Ruby version* (2002). 2022

Esse uso pode não ser vantajoso para casos em que o jogo não tenha uma mecânica que se beneficie do recurso, apesar de *A scientifically tested game* ser um RPG, não há nenhum aspecto do gameplay que necessite da contagem de tiles. Os combates do jogo não são iniciados da mesma maneira que em Pokémon. Eles se iniciaram ao encontrar um inimigo no cenário, como no jogo *Persona 5* (2016) feito pela desenvolvedora japonesa Atlus.



Figura 1: Inimigo avistado em Persona 5. Print Screen do jogo *Persona 5* (2016) . 2022

Atualmente, *tiles* ainda são usados em jogos 2D por ser uma estrutura de trabalho já conhecida e já presente em *engines* como *unity*, é algo consolidado. Apesar de jogos 2D não serem algo muito difícil de ser processado por computadores atuais, os *tiles* ainda sim, são mais eficientes para o processamento da máquina. Eles também tornam mais fácil programar algumas mecânicas do jogo, como a exemplificada em *Pokémon*. Os *tiles* também simplificam a criação de artes para o jogo, não é necessário desenhar uma cena inteira à mão, essa simplificação ocorre através da imposição de limites ao artista.

3.2 Problemas enfrentados no uso dos tiles

Cada *tile* do jogo foi composto por 16x16 pixels. Porém, à medida que o cenário era desenhado, percebia-se que adaptar as estruturas dos desenhos para o formato de *tile*, enquadrar os *props*⁹ dentro da resolução adequada e depois compor o cenário novamente dentro da engine, consumia mais tempo do que exportar as imagens já com os *props* agrupados e com eles em suas devidas posições, diretamente para a engine.

Cada cenário era desenhado no *Photoshop* de forma geral, para ter uma visão de como ficaria o cenário dentro do jogo em relação a proporção dos personagens, sendo as camadas separadas para cada *prop* do cenário, com o objetivo de posteriormente organizar o *Tile-set*. Porém, ao separar os *tiles*, as possibilidades de composição de cena foram restringidas na reorganização desses objetos dentro da *engine*, não tornando possível posicionar da mesma forma que havia sido idealizada inicialmente.

A imagem abaixo é um preview feito no photoshop de como o cenário se configuraria no jogo:

⁹ *Prop*: É um termo emprestado do teatro. É todo objeto usado pelos atores no palco em determinada peça. Em determinados textos, *props* em video-games são tudo aquilo que constitui uma cena mas não faz parte do layout do level ou do personagem, ou seja, qualquer objeto. Já outros autores dizem que o jogador deve ser capaz de interagir com determinado objeto para ele ser um *prop*. No caso deste artigo, usaremos a primeira definição, que são apenas objetos no cenário.



Figura 2: Concept do quarto do Egon. Imagem do autor. 2019

Na figura 3. traçamos uma grid que representa o *tile-map* sobre esse preview. Como podemos observar, vários objetos não se posicionaram de forma centralizada em cada *tile* e vários se sobrepuseram, como por exemplo a folha de papel ao lado do teclado, que acabou em uma interseção de tiles, sobrepondo o do teclado.

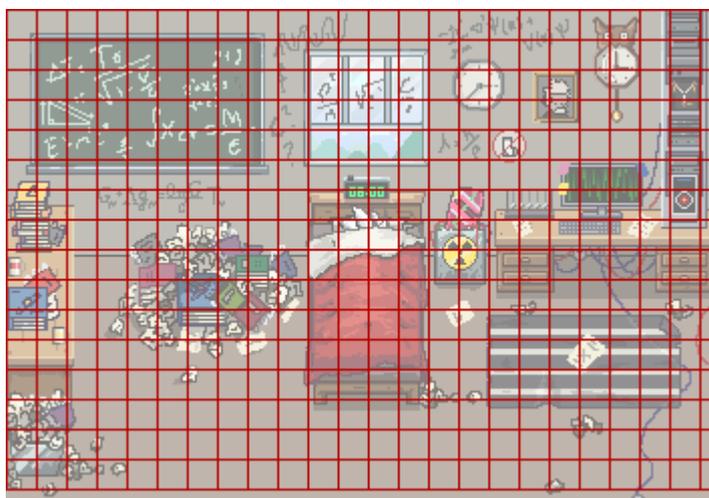


Figura 3: Grid sobre o quarto. Imagem do autor. 02 de junho 2022

Para preencher um *tile-map*, é necessário um *tile-set* e para fazer o *tile-set*, é necessário desenhar cada *prop* em uma camada diferente no *Photoshop*, para que seja mais fácil encontrar o *prop* no futuro caso seja necessário alguma alteração. Depois constituir uma imagem com todos os *props*, dispostos de uma maneira que se enquadrem nos *tiles* e não se sobreponham. Abaixo uma imagem do *tile-set*:



Figura 4: Primeiro *tile-set* quarto. Imagem do autor. 02 de junho 2019

O Photoshop foi o programa usado pois, pois nele era possível usar configurações específicas de pincéis das quais foram utilizados para criar diversas texturas no jogo, como de grama, na folhagem das árvores, em ladrilho do chão, nas paredes, etc. abaixo um exemplo de um cenário do jogo que apresenta essas texturas, na Figura 5 no piso das rampas que apresentam o desenho de losangos e na Figura 6 presente na grama:



Figura 5: Cenário da estufa de Dice. Imagem do autor. 2019



Figura 6: Imagem do exterior do laboratório de Kenzlyn. Imagem do autor. 2019

3.3 Alternativa para o uso de tiles

Como o jogo não teria nenhuma mecânica que dependesse do *tile-set*, surgiu a pergunta: por que fazer esse jogo com *tiles*? A partir dessa interrogação a ideia foi abandonada e os desenvolvedores do jogo passaram a separar os cenários em camadas: uma

camada em que o personagem controlado pelo jogador, fica à frente e outra em que os objetos ficam à frente do jogador, libertando os idealizadores do jogo das restrições impostas pela configuração do *Tile-map*.

No campo esquerdo da Figura 7, observa-se o cenário que pertence à camada em que o personagem controlado pelo jogador estará acima, em que o *sprite*¹⁰ do personagem controlado sobrepõe ao cenário. À direita, vê-se os objetos que tiveram de ser separados do cenário, já que eles estão em uma camada que sobrepõe a do personagem do jogador. No caso dos números, eles pertencem ao relógio e estão separados já que eles são animados e mudam as horas do relógio no decorrer do tempo.



Figura 7: Quarto do Egon final. Imagem do autor. 2019

Com este procedimento tivemos mais liberdade para texturizar a parede e o chão de maneiras diferentes, de forma a apresentar uma variação mais uniforme, sem fazer o uso repetitivo de um mesmo *tile*. Esse método não descarta a possibilidade de reutilizar itens do cenário em outros momentos do jogo, já que alguns dos objetos são separados por camadas no *photoshop*.

As próximas imagens exemplificam um cenário mais complexo. A figura 8 é um *preview*¹¹ do jogo. A figura 9 apresenta a camada dos objetos que ficam atrás do jogador e a 10, os objetos que ficam à frente do jogador em algum momento dentro da dinâmica do jogo.

¹⁰ *Sprite*: Imagens que representam tudo aquilo que é visualmente representado no jogo, nesse caso, o personagem e suas animações.

¹¹ Nesse contexto, ao utilizar a palavra *preview*, significa que é uma pré-visualização. A arte do cenário feita no *photoshop* pensando exatamente em como seria essa cena no jogo, como o jogador veria se estivesse jogando. Não é chamado de *concept* por causa do nível de refinamento e por não ser um desenho que explore o design.



Figura 8: Preview da rua. Imagem do autor. 2020



Figura 9: Background Rua. Imagem do autor. 2020



Figura 10: Layer 1 Rua. Imagem do autor. 2020

4. ROTEIRO

4.1 As convenções do roteiro para live actions

Quando o roteiro do Jogo começou a ser escrito, seu formato era baseado no que o autor e roteirista Syd Field fala em seu livro. Porém, Syd Field escrevia para filmes, não para

jogos eletrônicos, por isso, a formatação do roteiro tradicional não abrange tudo aquilo que acontece em um jogo, como por exemplo as diversas possibilidades de escolha que um jogador pode tomar no decorrer do gameplay.

Ao escrever um roteiro para *live action* ou animação de acordo com as convenções empregadas para histórias lineares, sempre existirá um início, um meio e um fim. É como Syd Field comenta em seu livro:

Porque um roteiro é uma história contada através de imagens, nós podemos nos perguntar, o que todas as histórias têm em comum? Elas tem um início, meio e fim, não necessariamente nessa ordem, como Jean-Luc Godard diz. Roteiros tem uma estrutura linear básica que cria a sua própria forma, segurando os elementos individuais e pedaços da história em seus devidos lugares. (Syd Field, 1979, p. 20)¹²

Porém, Syd Field não sabia o que os jogos eletrônicos iriam se tornar - que passariam a ter a sua própria estrutura narrativa. A forma de se escrever roteiro, descrita pelo autor, pode ser aplicada apenas parcialmente para os jogos. Os jogos que apresentarem apenas um desfecho são aqueles que podem se valer das orientações do autor. Em momentos isolados do jogo, como em *cutscenes*¹³, essa estrutura também é eficaz. No entanto, quase nenhum jogo é linear, mesmo que sua história seja, quase nenhum jogo será jogado duas vezes da mesma maneira. Até mesmo nas ações mais simples do jogador, como por exemplo, se ele abrir o jogo um dia e escolher andar para direita e no dia seguinte abrir o jogo no mesmo ponto e escolher andar pra baixo, o jogo já foi jogado de maneira diferente, por isso a estrutura do roteiro para jogos quase nunca é linear, em algumas situações essa escolha do jogador de andar pra baixo ou para direita pode iniciar consequências diferentes, ele pode encontrar um inimigo que só tem a direita por exemplo e morrer ou sobreviver para voltar para posteriormente, andar para baixo.

4.2 Soluções para como descrever no roteiro a imprevisibilidade dos video-games

Algumas das ações do jogador devem ser previstas pelo roteiro, caso elas tenham consequências específicas. Um exemplo é quando o jogador deve entrar em um local que está fechado. Ao chegar no local, o jogador tem a opção de ir direto para a porta da frente ou não

¹² Because a screenplay is a story told with pictures, we can ask ourselves, what do all stories have in common? They have a beginning, middle, and an end, not necessarily in that order, as Jean-Luc Godard says. Screenplays have a basic linear structure that creates the form of the screenplay because it holds all the individual elements, or pieces, of the story line in place. (tradução nossa)

¹³ *Cutscene*: É um filme que passa para o jogador em alguns momentos do jogo com o intuito de contar a história.

ir nesta direção. Se ele for primeiro até a porta da frente, o personagem que o acompanha irá dizer: “A porta está trancada, precisamos encontrar outra maneira de entrar”. Mas caso ele entre pela lateral e encontre o caminho real - que é subir a escada e pular a janela - nós podemos restringir o jogador e programar o outro personagem para dizer: “Vamos tentar a porta da frente primeiro”, ou permitir que ele pule a janela e não ative o diálogo. Esse foi um problema enfrentado durante o processo de criação do jogo: como organizar essas diversas possibilidades dentro de um roteiro e em quais momentos restringir o jogador.

O jogo também pode apresentar ramificações de possibilidades inseridas na própria história, com finais diferentes, ou até mesmo, um “meio” diferente. A única estrutura fixa da história, é o início. Como o nosso jogo na verdade, é uma *demo*, que abrange o prólogo de sua história (o início), não há porque ter mais de um final. Sendo assim, na tentativa de descrever os diversos caminhos a serem percorridos pelo jogador, o roteiro do jogo foi organizado da seguinte forma:

Uder e Egon(jogador) caminham para cima em direção a porta. Ao interagir com ela:

OPÇÃO 1: O jogador opta por explorar a lateral esquerda do cenário antes de interagir com a porta.

UDER

Você não acha mais fácil tentar a porta da frente primeiro?

Segue para a OPÇÃO 2.

OPÇÃO 2: O jogador interage primeiro com a porta da frente

EGON

A porta está trancada.

As luzes começam a piscar nas janelas.

Esse trecho acima é o que poderia ser chamado segundo a escritora para jogos AAA¹⁴, Susan O'Connor de *in-game script*. É a parte do roteiro mais volátil já que ela está sujeita ao *feedback* das ações do jogador de forma direta. Susan também menciona que não há um padrão para se escrever roteiros para vídeo-games, então essa foi a forma encontrada pela equipe idealizadora de *A scientifically tested game*: separar o roteiro em opções e caso uma tenha como consequência uma outra opção, apenas especificar isso ao final da sequência. Em uma situação hipotética em que existam cinco opções, se o jogador escolher a opção 1, ele pode ter no seu fim, como reação dessa escolha, a opção 4 por exemplo e assim por diante (O'CONNOR, 2020).

5. DESIGN DE PERSONAGENS

Por se tratar de um jogo *top-down* em *pixel art*, *A scientifically tested game* se torna esteticamente limitado à baixa resolução de pixels. Ao criar o rosto dos personagens isso se torna uma desvantagem, já que a resolução não permite explorar tão bem as expressões faciais. Por consequência, algumas nuances de suas personalidades não podem ser tão bem expressas, como o egocentrismo de Egon ou a seriedade de Dice - características essas que humanizam os personagens e os aproximam do jogador que se identifica com eles.

Em *Persona 5 (2016)*, os modelos 3D dos personagens também são limitados pela contagem poligonal dos modelos e não expressam bem, determinadas emoções e características. Desta forma, seus designers optaram pelo uso de retratos no estilo 2D nas caixas de diálogo de cada personagem que expressam com mais potência suas personalidades e emoções¹⁵. Sendo assim, adotamos essa mesma solução em *A scientifically tested game*, desenvolvemos retratos com uma maior resolução para as caixas de diálogo dos personagens.

¹⁴ Jogos AAA são jogos com maior investimento financeiro, feito por empresas de grande porte.

¹⁵ Vários outros jogos fazem uso de retratos mais detalhados em caixas de diálogo ou na seleção de personagens para poder dar maior expressividade e humanizá-los, como por exemplo *the Secret of the Monkey Island (1990)*, *Metal Slug 3 (2000)*, *Undertale (2015)*, *Stardew Valley (2016)*, *Hades (2018)*. O nível de resolução em pixels desses retratos pode variar, alguns ainda mantêm o visual de pixel art, outros não. Mas ainda sim, o nível de detalhe dos rostos é sempre maior que o rosto do personagem jogável.



Figura 11: Da esquerda para a direita: Victor, Dice, Uder, Egon e Kenzlyn. Imagem do autor. 2021

Os retratos evidenciaram a personalidade de cada personagem trazendo complexidade para seus designs. Explicaremos a seguir, as características que esses retratos permitiram agregar ao design dos personagens. Abaixo uma imagem retirada do jogo de como funcionam os balões de diálogo:



Figura 12: Diálogo entre Egon e Uder em frente ao laboratório de Kenzlyn. Imagem do autor. 2022

5.1 Egon

Através do retrato foi possível deixar mais evidente a personalidade rebelde de Egon. O ângulo em que ele é retratado, levemente visto de baixo, expressa seu ar de superioridade e egocentrismo - sinais que evidenciam seu sentimento de superioridade com relação aos

demais integrantes do grupo. Como lembra Katherine Isbister, no geral, pessoas dominantes tendem a se posicionar fisicamente no espaço de maneira mais elevada que outras pessoas (ISBISTER, 2006).

O sorriso e a sobrancelha também acentuam sua superioridade e também contribuem para seu ar de sarcasmo e perspicácia. O cabelo é branco no intuito de evidenciar que ele é um jovem rebelde. Em mangás e animes japoneses, os jovens de cabelo descolorido sempre possuem essa conotação. Como o personagem *Ryuji Sakamoto* em *Persona 5* e o personagem *Gohan* quando mantém a forma de *Super Sayajin* em *Dragon Ball Z* no episódio 169. O cabelo branco também segue o estereótipo de cientista louco, numa analogia ao físico Albert Einstein, já que no jogo a especialidade desse personagem é o campo da relatividade.

Fazemos o uso do estereótipo como vantagem, porém, alguns traços de Egon desconstruem esse estereótipo ao mesmo tempo. Esta contradição é colocada como positiva por Katherine Isbister: “Bons designers também constroem personagens memoráveis pegando estereótipos famosos e introduzindo alguns traços que se desvencilham desse mesmo estereótipo” (IBISTER, 2006). O maxilar mais definido do personagem, assim como o pescoço robusto o distanciam da imagem de cientista de corpo frágil que usa apenas a mente como recurso. O visual de Ego fortalece o caráter heroico sugerindo que ele tem personalidade forte.

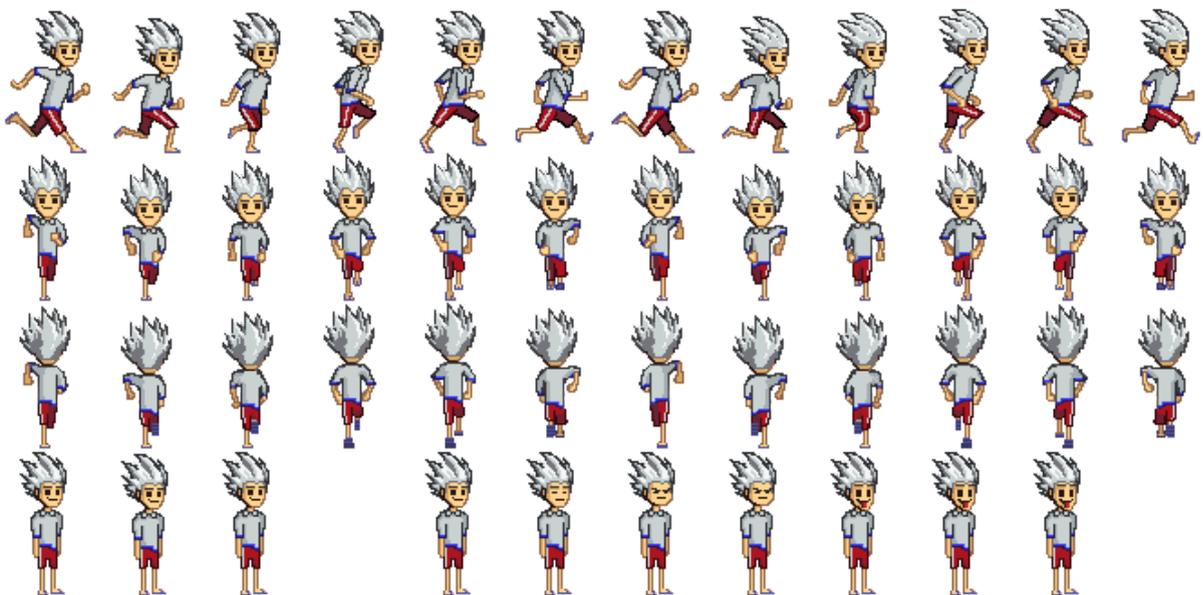


Figura 13: ciclo de caminhada do Egon. Imagem do autor. 2019

5.2 Dice

Dice é quase como um rival de Egon. Ele é inteligente como o protagonista, porém sua área de pesquisa é outra: mecânica quântica. No mundo real, Albert Einstein não concordava com as questões probabilísticas da mecânica quântica e como Egon é inspirado em Einstein, ele discorda das ideias de Dice e até mesmo do nome do personagem que é uma referência a uma carta escrita por Einstein em que ele menciona que Deus não joga dados. Como os dois personagens não se entendem a princípio, o retrato de Dice o representa de costas, olhando por cima dos ombros com seriedade e aspecto suspeito, evidenciando seu desejo em não estabelecer vínculo com o protagonista, conforme explica Katherine Isbister:

Como uma pessoa comunica amizade e está disposta a conversar? Ela faz isso com seu rosto, corpo, voz e com o seu padrão de comportamento geral em relação a outra pessoa. Todos esses sinais trabalham em conjunto para criar uma impressão geral sobre a pessoa que está sendo abordada. (ISBISTER, 2006)¹⁶

Por isso Dice é retratado como se ele não quisesse fazer contato visual com o protagonista: com óculos escuros e chapéu, se distanciando não só do protagonista como também dos demais jogadores. O objetivo é dar a sensação que ele tem potencial para se tornar um vilão e distrair o olhar do jogador do verdadeiro vilão, que é Victor. Dice, posteriormente se torna um dos melhores amigos e companheiros do protagonista, conforme Egon vai deixando de lado seu ego, isso vai revelando ao jogador que a atitude hostil de Dice era na verdade uma consequência das atitudes de Egon e que Dice na verdade, pode ser um grande aliado.

5.3 Uder

Uder, é o melhor amigo de Egon, ele é como a voz da consciência do protagonista, em determinadas situações quando o orgulho de Egon o impede de falar com alguém ou fazer algo, é Uder quem soluciona esse problema.

O melhor amigo do herói ou irmão, que nunca sai do lado do herói. Às vezes entusiasmado, às vezes relutante, ele providencia um balanço para o herói, às vezes o motivando a fazer algo ou então o impedindo,

¹⁶ How does a person telegraph friendliness and willingness to engage? They do so with face, body, and voice, as well as the overall pattern of behavior toward the other person. All of these signals work together to create a general impression in the person who is being approached. (tradução nossa)

fazendo com que ele reconsidere suas ações. (LEBOWITZ; KLUG, 2011)¹⁷

A característica relutante desse personagem está presente no seu olhar de canto de olho e expressão mais neutra, é como se ele estivesse constantemente preparado para resolver algum problema que seu amigo Egon provocou. Ele é o arquétipo do melhor amigo do herói com mais consistência nos pensamentos e atitudes. Mas como apontam LEBOWITZ e KLUG, um personagem não precisa seguir um arquétipo por todo o decorrer da história e por isso, eventualmente, Uder sai de cena e quem entra no lugar como melhor amigo é Dice.



Figura 14: ciclo de caminhada do Uder. Imagem do autor. 2019

5.4 *Kenzlyn e Victor*

Kenzlyn e Victor são irmãos gêmeos, por isso, foi usada a mesma base (cabeça, corpo, nariz, etc) de desenho para ambos, mudando os acessórios e suas expressões faciais. Kenzlyn é inocente, brincalhão e apresenta gostos peculiares. Por isso, o sorriso estampado na sua cara e o estilo diferente, com um bigode e um mullet. “Um rosto amigável irá provavelmente ter um sorriso em sua face ou pelo menos uma impressão disso em seus olhos” (IBISTER, 2006)¹⁸. Como o campo da física em que esse personagem é especialista é o

¹⁷ The hero's best friend or sibling, she never leaves the hero's side. Sometimes enthusiastic, sometimes reluctant, she provides a balance for the hero, either pushing him to action or urging him to slow down and carefully consider his actions. (tradução nossa)

¹⁸ A friendly person will probably have a smile on their face or at least the hint of a smile in their eyes. (tradução nossa)

eletromagnetismo, ele apresenta sobranceiras com formatos que lembram os de raios e o brilho em seu cabelo cumpre a mesma função.

Já Victor é totalmente o oposto, ele é ranzinza, e detesta as atitudes egoístas de Egon. Por isso, ele é retratado com um semblante sério, como se estivesse sendo visto de cima. Victor, eventualmente se torna o vilão e o antagonista de Egon. Para não tornar isso muito óbvio, já que desde o início os dois não se entendem, foi feito o design de Dice de forma que a atenção saísse de Victor e fosse para ele.

6. CONSIDERAÇÕES FINAIS

O objetivo deste trabalho foi expor os problemas enfrentados durante a produção de um jogo 2D e mostrar as soluções encontradas pela equipe a fim de sugerir e demonstrar novas perspectivas para outras pessoas que queiram se aventurar no processo de criação de jogos. O foco foram os aspectos artísticos dos quais este autor foi responsável.

Foi feita então uma análise dos seguintes aspectos do jogo: *Engine*, Roteiro e Design de personagem. Em que na engine foi abordado o conceito de *tile-map* e *tile-set* e como o uso desses recursos apesar de serem muito utilizados na produção de jogos, também podem apresentar defeitos de acordo com as especificidades de cada jogo. É necessário então se perguntar se as mecânicas do jogo irão se beneficiar desse recurso e ter um propósito, uma justificativa para seu uso.

No caso da demo de *A scientifically tested game*, o uso desse recurso se mostrou desnecessário, já que o jogo não apresenta nenhuma mecânica que necessite do uso de tiles. Eles podem ser sim, muito eficientes em outros contextos mas também são por definição restritivos, além do que, organizar os sprites do jogo de forma a se enquadrar nos tiles é algo que consome muito tempo. Portanto, se o jogo não tiver nenhum aspecto de sua gameplay que necessite deste recurso, não há motivo para usá-lo. Os computadores atuais conseguem processar sem problemas um jogo que não faz uso de tiles. Por isso, apenas o aspecto da performance não justificaria seu uso, e isso não ocorre apenas por que *A scientifically tested game* é uma demo, um jogo “menor”, pois *Braid* (2009), um jogo indie projetado por Jonathan Blow, foi inteiramente criado sem o uso de tiles.

Em roteiro, defendemos que o formato padrão da escrita de roteiros para filmes não é eficaz quando direcionado para jogos, já que ele não engloba a imprevisibilidade e mutabilidade de um jogo. É necessário organizar o roteiro de forma ramificada, prevendo alguns dos possíveis caminhos que o jogador irá escolher, não apenas em termos de enredo, mas também de gameplay. Em *A scientifically tested game* vimos como exemplo, uma

situação em que a ação do jogador pode resultar em duas consequências diferentes, não sendo algo que afete diretamente a história mas sim a forma com que os personagens interagem entre si, através dos diálogos. É através desses diálogos em paralelo com a *gameplay* que algumas informações são reveladas ao jogador e por isso, a importância de ter no roteiro, diferentes respostas para as ações de quem está no controle do personagem. Vimos um exemplo de como evidenciar no roteiro, as diversas opções que o jogador possa vir a escolher.

Em design de personagem percebemos que a resolução original dos personagens jogáveis, enquanto *gameplay*, não era o suficiente para poder expressar algumas características da personalidade desses personagens. Uma forma de solucionar esta questão foi através dos retratos que aparecem com a caixa de diálogo, que não só indicam quem está falando como, também humaniza o personagem e os tornam mais íntimos do jogador. Vimos diversos exemplos de características que são expressadas, não apenas através do design, como também do uso de cores e principalmente, na maneira que o personagem é apresentado a expressão facial, a posição de seu corpo e o ângulo em que ele é retratado. Por isso, os retratos com uma maior resolução de pixels é tão importante, já que expressões faciais e corporais mais sutis são impossíveis de serem desenhadas com o limite de pixels baixo que temos nos sprites do personagem jogável.

Por fim, esse artigo apresenta o exemplo de um jogo que não faz uso de tiles e tem justamente o intuito de mostrar que, caso algum game designer tenha a intenção de utilizá-lo, isso é possível sendo que existem jogos que já funcionaram e utilizaram esse método.

Acreditamos que o artigo apresentou alternativas para alguns aspectos da organização de roteiros para vídeo-games como também soluções para conferir maior expressividade no design de personagens em jogos de pixel-art. Esperando assim poder contribuir em futuros processos de game designers que venham a dar seus primeiros passos na criação de jogos.

7. REFERÊNCIAS

- BAGGOTT, Jim. **What Einstein meant by ‘God does not play dice’**. Disponível em: <<https://www.britannica.com/story/what-einstein-meant-by-god-does-not-play-dice#:~:text=This%20article%20was%20originally%20published,>>. Acesso em: 06 de julho de 2022.
- DAVIES, Paul. **Spider-man: The Art of the Game**. Londres: Titan Books, 2018.
- FAMITSU. **Persona 5: Official Design Artworks**. Japão: Kadokawa Shoten, 2016.
- FIELD, Syd. **Screenplay: The Foundations of Screenwriting**. Edição Revisada. Nova York: Bantam Dell, 2005.
- ISBISTER, Katherine. **Better Game Characters by Design: A Psychological Approach**. San Francisco: Morgan Kaufmann, 2006.
- J.P. WOLF, Mark. **Before the Crash: Early Video Game History**. Detroit: Wayne State University Press, 2012.
- LEBOWITZ, Josiah; KLUG, Chris. **Interactive Storytelling for Video Games**. Elsevier 2011.
- O’CONNOR, Susan. **How to write a script for games**. Disponível em: <https://www.susanoconnorwriter.com/blog/script-writing-for-games>. Acesso em: 04 de Julho de 2022.
- SHELDON, Lee. **Character Development and Storytelling for Games**. Boston: Premier Press, 2004.
- The implementation of rewind in Braid. GDC. **Youtube**. 12 dez. 2016. 26min55s. Disponível em: <https://www.youtube.com/watch?v=8dinUbg2h70&ab_channel=GDC>. Acesso em: 06 de julho de 2022.
- Are Unity’s Tile Features Good Enough to make real games?. AdamCYounis. **Youtube**. 24 Julho. 2021. 36min19s. Disponível em: <https://www.youtube.com/watch?v=wRkT10D-Quo&t=1584s&ab_channel=AdamCYounis>. Acesso em: 09 de julho de 2022.