



UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO DE COMUNICAÇÃO E EXPRESSÃO
DEPARTAMENTO GMT
CURSO ANIMAÇÃO

Rodrigo Werle Rezende

Promovendo a universidade através dos Serious Games: Protótipo de um jogo
utilizando ferramentas gratuitas

Florianópolis
2023

Rodrigo Werle Rezende

Promovendo a universidade através de Serious Games: Protótipo de um jogo
utilizando ferramentas gratuitas

Trabalho de Conclusão de Curso submetido ao curso de Animação do Centro de Comunicação e Expressão da Universidade Federal de Santa Catarina para a obtenção do título de Bacharel em Animação.

Orientador(a): Dr. Nicholas Brugger Grassi

Florianópolis

2023

Rezende, Rodrigo Werle

Promovendo a universidade através dos serious games :
Protótipo de um jogo utilizando ferramentas gratuitas / Rodrigo
Werle Rezende ; orientador, Nicholas Bruggner Grassi, 2023.
63 p.

Trabalho de Conclusão de Curso (graduação) - Universidade
Federal de Santa Catarina, Centro de Comunicação e Expressão,
Graduação em Animação, Florianópolis, 2023.

Inclui referências.

1. Animação. 2. Serious games. 3. Gamificação. 4.
Desenvolvimento de jogos. 5. 3D. I. Grassi, Nicholas Bruggner .
II. Universidade Federal de Santa Catarina. Graduação em
Animação. III. Título.

Rodrigo Werle Rezende

Promovendo a universidade através dos Serious Games: Protótipo de um jogo utilizando ferramentas gratuitas

Este Trabalho de Conclusão de Curso (TCC) foi julgado adequado para obtenção do Título de Bacharel em Animação e aprovado em sua forma final pelo Curso de Animação da Universidade Federal de Santa Catarina.

Florianópolis, 19 de Junho de 2023.

Prof. Flávio Andaló, Dr. Coordenador do Curso de Animação UFSC

Banca Examinadora:

Nicholas Bruggner Grassi, Doutor. Universidade Federal de Santa Catarina

Flávio Andaló, Doutor. Universidade Federal de Santa Catarina

Gabriel de Souza Prim, Doutor. Universidade Federal de Santa Catarina

Professor/a Orientador/a
Universidade Federal de Santa Catarina

RESUMO

Os jogos eletrônicos tornaram-se uma forma de entretenimento popular em todo o mundo, inclusive no Brasil. Nesse contexto, um protótipo de jogo foi desenvolvido, com o objetivo de explorar o ambiente universitário como tema principal. Um aspecto importante desse projeto foi o uso exclusivo de softwares gratuitos disponíveis no mercado. Durante o processo de desenvolvimento, foram analisados exemplos reais de jogos educacionais que visam promover a aprendizagem através do conceito da gamificação.

Palavras-chave: Gamificação; Serious games; Desenvolvimento de jogos; Exploração; 3D.

ABSTRACT

Electronic games became an incredibly popular form of entertainment worldwide, including in Brazil. In this context, a game prototype was developed with the aim of exploring the university environment as its main theme. An important aspect of this project was the exclusive use of free software available in the market. During the development process, real examples of educational games that aim to promote learning through the concept of gamification were analyzed.

Keywords: Gamification; Serious Game; Game development; Exploration; 3D.

LISTA DE FIGURAS

Figura 1 - Gráfico da pesquisa sobre o consumo de jogos.....	18
Figura 2 - Gráfico de faturamento global no mercado de jogos do ano de 2022.....	19
Figura 3 - Telas do jogo Capture The Museum.....	22
Figura 4 - Jogo educativo disponível na plataforma Google Arts & Culture.....	23
Figura 5 - Exemplo visual do sistema de Nodes de uma Blueprint.....	26
Figura 6 - Exemplo de um objeto que possui vértices, arestas e faces.....	27
Figura 7 - Processo de Box Modeling.....	28
Figura 8 - Exemplo de como o Uv Map e a malha funcionam na textura.....	30
Figura 9 - Exemplo de um Rig.....	31
Figura 10 - Exemplo do modelo de produção.....	33
Figura 11 - Referências de jogos.....	34
Figura 12 - Exemplo de personagens com Cell Shading.....	36
Figura 13 - Painel de referência visual.....	37
Figura 14 - Fotos de referência da universidade.....	38
Figura 15 - Etapa de blocagem do cenário no software Blender.....	39
Figura 16 - Primeiro teste de importação do cenário para a Unreal.....	40
Figura 17 - Blocagem do primeiro andar do prédio.....	41
Figura 18 - Blocagem inicial dos objetos no cenário.....	42
Figura 19 - Processo de blocagem no Blender.....	43
Figura 20 - Blocagem cenário Design Lab.....	44
Figura 21 - Design Lab.....	45
Figura 22 - Modelagem de asset.....	46
Figura 23 - Textura do mural presente em um dos corredores.....	47
Figura 24 - Painel de materiais da parede na Unreal.....	48
Figura 25 - Primeira versão jogável do protótipo.....	49
Figura 26 - Segunda versão jogável do protótipo.....	49
Figura 27 - Modelagem da terceira versão do personagem.....	50
Figura 28 - Início da modelagem da quarta versão do personagem.....	51
Figura 29 - Processo de modelagem de personagem.....	51
Figura 30 - Rigging personagem.....	52
Figura 31 - UV Map personagem.....	53
Figura 32 - Texturização do personagem utilizando o software Krita.....	54
Figura 33 - Referência de ciclo de caminhada para a animação.....	55
Figura 34 - Blueprint do personagem Ufscão na Unreal.....	56
Figura 35 - Tela de Animation Blueprint.....	56
Figura 36 - Tela de Blend Space.....	57
Figura 37 - Mascote do curso de animação UFSC.....	58
Figura 38 - Processo de retopologia do personagem mascote.....	59
Figura 39 - Processo de ajuste de weight paint.....	59
Figura 40 - Uv Map e textura do personagem mascote.....	60
Figura 41 - Personagem Ufscão.....	61
Figura 42 - Personagem mascote.....	61

Figura 43 - Bloco A, CCE, Térreo.....	62
Figura 45 - Bloco A, CCE, 1º andar.....	63
Figura 46 - Bloco A, CCE, 1º andar em jogo.....	64
Figura 47 - Bloco A, CCE, 1º andar, parede.....	64
Figura 48 - Bloco A, CCE, 1 andarº, parede em jogo.....	65
Figura 49 - Sala Design Lab.....	65
Figura 50 - Sala Design Lab em jogo.....	66
Figura 51 - Sala 130.....	67
Figura 52 - Sala 130 em jogo.....	67

SUMÁRIO

1 INTRODUÇÃO	16
1.1 OBJETIVOS	16
1.2 PROBLEMÁTICA	17
1.3 JUSTIFICATIVA	17
2 FUNDAMENTAÇÃO TEÓRICA	19
2.1 SERIOUS GAMES E GAMIFICAÇÃO	20
2.2 MOTOR GRÁFICO	24
2.3 Modelo 3D	26
2.4 Modelagem em Box Modeling	27
2.5 Topologia	28
2.6 Texturização	29
2.7 Animação 3D	30
3 Desenvolvimento	32
3.1 Pré-produção	34
3.2 Produção	37
3.2.1 Blocagem do Cenário	37
3.2.2 CICLO DE PRODUÇÃO DOS ASSETS	45
3.2.2.1 MODELAGEM DOS ASSETS	46
3.2.2.2 MATERIAIS E TEXTURA DOS ASSETS	47
3.2.3 PRODUÇÃO DE PERSONAGENS	48
3.2.3.1 UFSCÃO	48
3.2.3.2 NPC	57
3.2.3.3 MASCOTE	58
5 CONSIDERAÇÕES FINAIS	68
REFERÊNCIAS	69

1 INTRODUÇÃO

Este artigo é referente às etapas do processo de criação de um *Serious Game*, jogo eletrônico com ênfase na exploração e familiarização de espaços urbanos tendo como foco a UFSC (Universidade Federal de Santa Catarina). Assim, foram abordados diferentes tipos de jogos disponíveis no mercado, aplicativos que têm algum propósito de simulação, a existência de *softwares* que permitem explorar o mundo sem precisar sair de casa e questões necessárias para a produção de um jogo.

A ideia do projeto é a produção de um jogo que tenha como foco a Universidade Federal de Santa Catarina, em que o jogador controlará um cachorro e irá sair pelo campus conhecendo diversos personagens que possuem relação com a universidade.

Devido ao cronograma limitado e à complexidade envolvida na criação de jogos, o produto final produzido para este trabalho foi um protótipo contendo 2 personagens: um cão e o mascote.

1.1 OBJETIVOS

O objetivo deste projeto foi desenvolver um protótipo de jogo em 3D, com o foco em exploração, utilizando apenas ferramentas gratuitas como *Unreal Engine*, *Blender* e *Krita*, tendo como o principal intuito uma apresentação gamificada da Universidade Federal de Santa Catarina. Para isso, buscou-se:

- (1) Pesquisa sobre o conceito dos serious games e os benefícios do uso da gamificação para a apresentação de conteúdos;
- (2) Pesquisa do uso real dessa mídia tanto em locais físicos tais quais museus, como em ambientes virtuais como sites e aplicativos;
- (3) Pesquisa de *pipelines* de desenvolvimento de jogos;
- (4) Aplicação do que foi investigado no para o desenvolvimento do trabalho, descrevendo as diversas habilidades que foram necessárias para a produção do protótipo, tendo em vista o objetivo final pré-estabelecido no *GDD*.

1.2 PROBLEMÁTICA

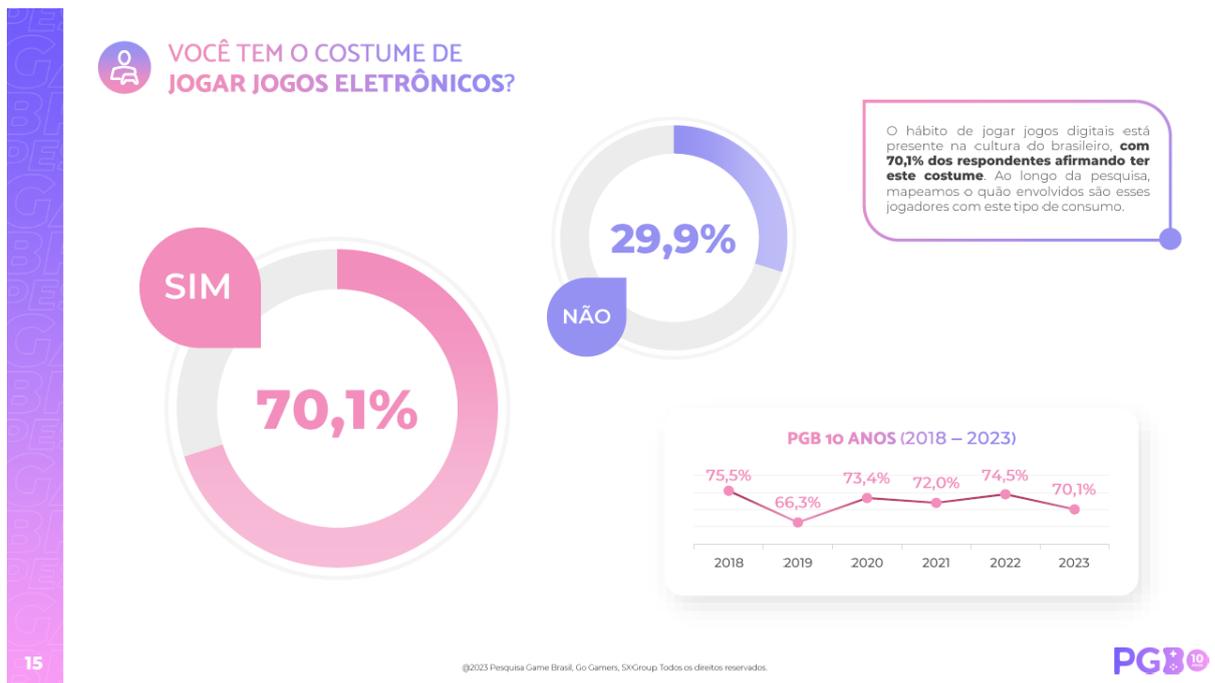
A UFSC é uma universidade pública com muito a oferecer, tanto em espaços físicos quanto em instalações que podem ser utilizadas por membros da universidade e pela comunidade, além de fomentar a produção de uma ampla variedade de projetos e pesquisas. No entanto, muitas vezes a informação sobre esses recursos pode não chegar adequadamente às pessoas, especialmente para aquelas que não têm vínculo com a universidade e que não convivem diretamente nesses espaços. Sendo assim, soluções digitais que facilitem o acesso a essas informações de forma mais engajadora podem contribuir nesse contexto.

1.3 JUSTIFICATIVA

O desenvolvimento de jogos tem crescido de forma significativa em todo o mundo, e o Brasil não é exceção. Com o aumento do interesse por jogos eletrônicos e a crescente demanda por entretenimento digital, o mercado de jogos no Brasil tem experimentado um aumento constante nos últimos anos. Isso tem levado a um aumento no número de empresas de desenvolvimento de jogos no país. Segundo uma pesquisa da Abragames (2022), o número de desenvolvedoras de jogos no Brasil cresceu significativamente ao longo dos anos. Em 2014, foram registradas 144 desenvolvedoras, em 2021 esse número subiu 1009, um aumento de mais de 600%. Além disso, a pesquisa revelou que 74,5% da população brasileira jogam videogames, tendo um foco expressivo em jogos mobile

Outra pesquisa que reforça essa informação vem da PESQUISA GAME BRASIL (2023), em uma pesquisa contendo uma amostra de 14.825 pessoas de variadas faixas etárias, mostrou que 70.1% dos brasileiros possuem o costume de jogar videogames, e desses, 82.1% afirmam que essa é uma de suas principais formas de diversão. É possível observar o gráfico dessa informação na Figura 1.

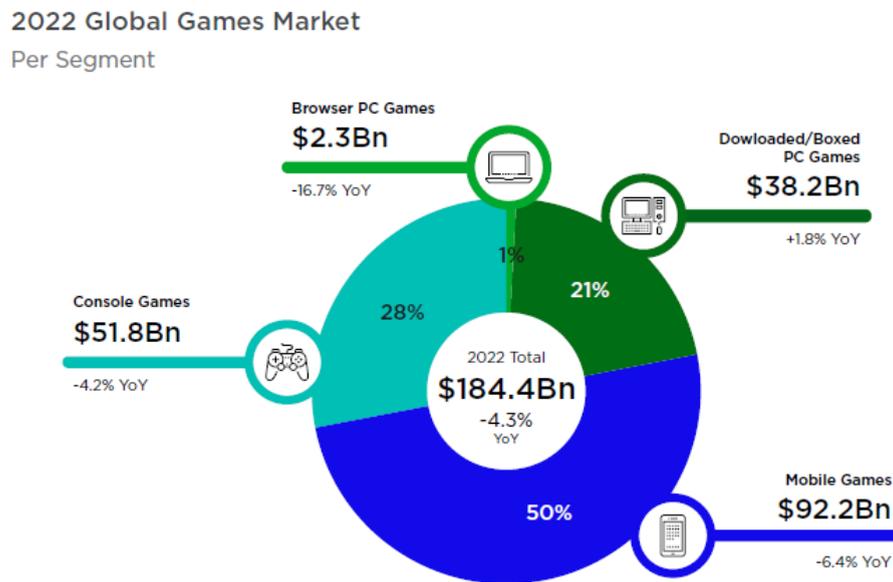
Figura 1 - Gráfico da pesquisa sobre o consumo de jogos



Fonte: Pesquisa Game Brasil, 2023.

Essa grande popularidade dos jogos eletrônicos também podem ser averiguados pelo faturamento das empresas do setor, segundo a pesquisa *Global Games Market Report* realizada pela Newzoo (2022), os jogos eletrônicos geraram 184.4 bilhões de dólares em receita globalmente (Figura 2), com um destaque ao mercado mobile, que foi responsável por 50% desse faturamento. Além disso, a pesquisa mostra que as projeções estimam que em 2025 o mercado irá faturar um total de 211.2 bilhões de dólares. Outra pesquisa da Newzoo (2022), a *Key Insights Into Brazilian Gamers*, aponta que o Brasil foi o 10º país que mais faturou com jogos no ano de 2022 com um total de \$2.7 bilhões de dólares, e foi o também o país com a 5ª maior quantidade de jogadores, tendo um total de 101 milhões de pessoas que jogam videogames regularmente.

Figura 2 - Gráfico de faturamento global no mercado de jogos do ano de 2022



Fonte : Newzoo, 2022.

A partir das informações levantadas por essas pesquisas recentes que refletem o interesse do público brasileiro em jogos, em um país que possui o 5º maior público de jogadores e apenas é o 10º em questão de faturamento é interessante um maior fomento para o desenvolvimento de um mercado forte no país, acredita-se que a linguagem dos games possa agregar valor a outras propostas que também buscam o engajamento.

2 FUNDAMENTAÇÃO TEÓRICA

Para a produção deste documento, foi necessário realizar uma pesquisa acerca da área de serious games e da produção de jogos usando como base materiais já existentes relacionados ao tema do projeto a fim de fornecer um maior conhecimento na área de estudo.

2.1 *SERIOUS GAMES E GAMIFICAÇÃO*

Por mais que os jogos tenham majoritariamente um caráter de entretenimento, existem alguns que possuem uma função além da diversão. Esses jogos são conhecidos como *Serious Games*, e têm se mostrado uma poderosa ferramenta de aprendizado e conscientização em diversas áreas, desde a educação infantil até o treinamento corporativo (Michael; Chen, 2009). É possível compreender o *Serious Game* como um jogo em que a educação é o objetivo primário ao invés do entretenimento (Michael; Chen, 2009), ou seja, esses jogos tem como finalidade ensinar ou instruir o usuário sobre algum assunto utilizando jogos como mídia para tal objetivo.

Ao desenvolver *serious games*, o foco está em combinar o conteúdo do jogo com o objetivo pretendido ao criá-lo; o jogo deve, de fato, alcançar um objetivo além do entretenimento. (Amanatiadou e Van de Weerd, 2009, tradução livre).

Jogos causam interesse do usuário por sua interatividade, ou seja, o usuário não recebe apenas a informação, mas sim participa do processo de construção dela de uma maneira mais próxima, sem serem forçados e estando envolvidos com o *game*, aumentando a probabilidade de aprenderem (MATTAR, 2010, p.17).

Um ponto importante dos *serious games* é o fator conhecimento, e devem ser projetados para serem, além de divertidos e envolventes, responsáveis por um aprendizado. As gerações mais jovens cresceram em meio aos videogames, uma forma de mídia com a qual estão familiarizados desde cedo, dessa forma, eles têm uma maior propensão a jogar e aprender por meio dessa experiência. (Michael; Chen, 2009, p.26).

Além disso, por meio da gamificação, ou seja, da utilização de elementos e mecânicas de jogos em contextos do cotidiano, é possível engajar o público de forma mais efetiva e aumentar a retenção de informações (ALVES, 2015). A gamificação de espaços públicos utilizando *serious games* para dar uma abordagem lúdica de interação pode ser uma solução envolvente para atrair visitantes a locais como museus, por exemplo. De acordo com Pivec e Kronberger (2016), ao trazer elementos de gamificação para esses ambientes, é possível criar uma experiência divertida não apenas para indivíduos, mas também para famílias e

grupos, proporcionando novas oportunidades de engajamento no processo de aprendizagem. Alguns museus estão implementando elementos de gamificação em suas exposições para aumentar o engajamento do público e ganhar a atenção das gerações mais novas, proporcionando uma maneira diferente de interação com suas obras e oferecendo diferentes níveis de envolvimento.

O aplicativo do Rijksmuseum em Amsterdã oferece aos visitantes uma experiência lúdica por meio do recurso chamado *Family Quest*. Neste jogo, de dois a quatro jogadores podem participar, utilizando seus próprios dispositivos digitais enquanto exploram o museu. O desafio proposto é encontrar segredos ocultos nos objetos expostos no museu. No Museu Nacional da Escócia, em Edimburgo, é adotada uma abordagem semelhante por meio do jogo chamado *Capture the Museum* demonstrado na Figura 3. Nesse jogo em grupo, duas equipes têm a oportunidade de competir entre si em uma versão ao vivo, semelhante ao popular jogo *Capture the Flag*. O objetivo é capturar e manter diversos locais físicos dentro do museu para alcançar a vitória.

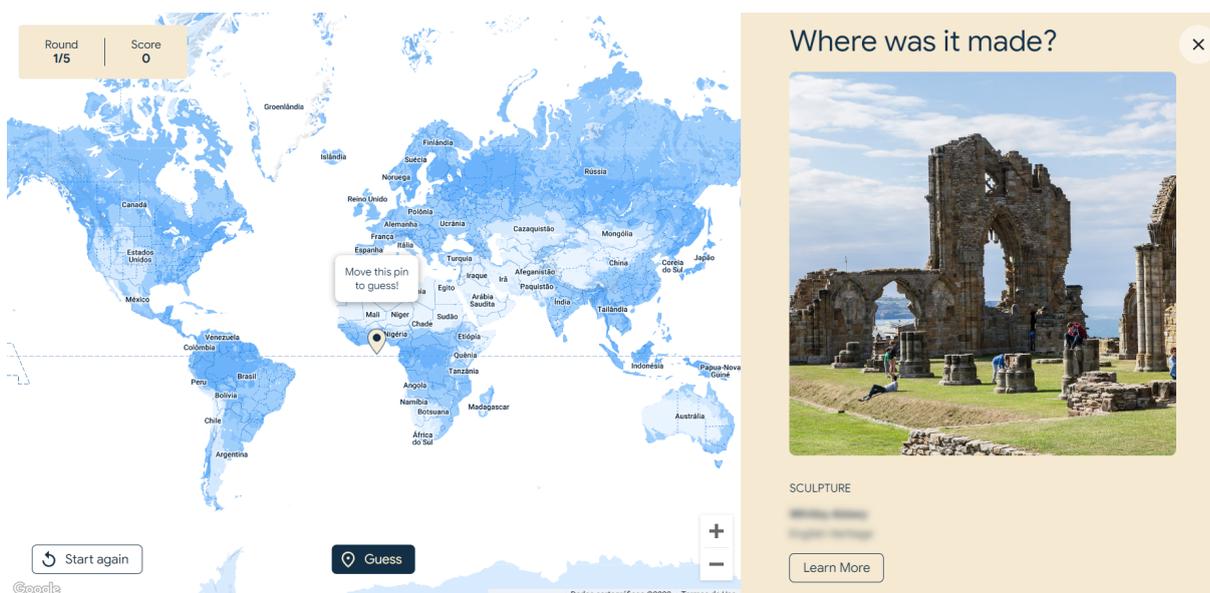
Essas abordagens promovem a interação e o engajamento dos visitantes, tornando as visitas mais envolventes e divertidas.

Figura 3 - Telas do jogo *Capture The Museum*



Outra abordagem interessante vem da empresa *Google*, o *Google Arts & Culture* (2011) que é um site sem fins lucrativos (Figura 4) visando a distribuição de cultura de uma maneira mais acessível, disponibilizando uma infinidade de coleções para a visualização digital em uma espécie de *google street view*¹. Isso permite que o usuário possa passear de maneira dinâmica pelo interior de diversos museus disponíveis e visualizar os acervos de uma forma mais interativa do que apenas olhar fotos estáticas. Além disso, o site também oferece diversos jogos que possuem um teor educativo envolvendo conteúdos como pintura, história e música, sendo exemplos do uso de *serious games*.

Figura 4 - Jogo educativo disponível na plataforma *Google Arts & Culture*



Fonte: <https://artsandculture.google.com/experiment/geo-artwork/wgEPVBAUIRVIEQ>

O desenvolvimento de jogos, sendo ele *Serious Game* ou não, envolve uma série de processos complexos que requerem conhecimentos abrangentes em várias disciplinas (ASLAN e BALCI, 2015). Assim, encontramos uma variedade de profissionais que desempenham papéis essenciais no processo de criação. Isso inclui programadores, engenheiros de *software* e desenvolvedores, designers, artistas, atores e dubladores, roteiristas, músicos, entre outros (NETO, 2017).

¹ O *Google Street View* é um aplicativo que permite ao usuário visualizar e transitar remotamente por espaços formados de imagens em 360 graus de vários locais do mundo.

Dessa forma, o jogo se torna uma forma de expressão artística complexa e multidisciplinar, que demanda o conhecimento e a colaboração de profissionais de diversas áreas para criar experiências imersivas e envolventes para os jogadores. Assim, vê-se que jogos vão além da expressão artística, da criação baseada em criatividade e pensamento imaginativo, eles representam uma habilidade que envolve diferentes disciplinas criada a partir da combinação de aspectos intercalados que se entrelaçam e se integram de forma multidisciplinar (Wardrip-Fruin e Montfort, 2003)

Existem empresas que desenvolvem jogos complexos, chamados de AAA (*Triple A*), que possuem um orçamento que muitas vezes alcançam centenas de milhões de dólares e possuem equipes que chegam a centenas de integrantes. Existem também grupos pequenos de desenvolvedores que possuem uma quantidade limitada de recursos para a produção de seus projetos, mas, como explica ROCHA (2015), existem também os desenvolvedores independentes que acabam não contando com nenhum auxílio além da própria equipe, como foi o caso deste projeto.

Com isso em mente, um fluxo de trabalho adequado para tal tarefa a fim de conseguir produzir um produto que possua a qualidade buscada dentro do escopo definido para o projeto.

Durante as pesquisas sobre as metodologias existentes para a produção de jogos, se constatou que há diversos modelos de pipelines, ou seja, sequência de etapas/processos. De forma geral, as pipelines pesquisadas são organizadas de forma muito similar, dividindo-se em pré-produção, produção e pós-produção. Segundo Rogers (2013), a pré-produção se inicia definindo o escopo do jogo e documentando tudo em um GDD (*Game Design Document*), um documento de texto, criado pelo game designer. O GDD abrange uma ampla variedade de elementos do jogo, incluindo estética, narrativa, mecânicas e outros aspectos relevantes. Sua finalidade principal é comunicar e orientar todos os participantes envolvidos no processo de desenvolvimento do jogo, fornecendo diretrizes claras e servindo como um guia abrangente, ele estabelece uma visão unificada do jogo e garante a coerência e a colaboração eficaz durante todo o processo de criação (MOTTA e JUNIOR, 2013). A etapa de produção é o momento em que são produzidos e implementados os assets, ou seja, elementos gráficos, sonoros ou de programação utilizados para a construção de jogos. Por fim, na pós-produção,

ocorre um refinamento da experiência desejada utilizando recursos do motor gráfico (ROGERS, 2013).

2.2 MOTOR GRÁFICO

Engine (Motor gráfico) é um *software* que é usado para a produção de jogos eletrônicos, possuindo um conjunto diverso de ferramentas e funcionalidades prontas para uso que permite o usuário fazer modificações de acordo com o necessário para o projeto, ao invés de precisar criar e programar do zero, permitindo que o fluxo de trabalho seja bem mais rápido e efetivo (SALUTES, 2022; XIMENES, 2022). Existem diversas engines no mercado que são utilizadas tanto para produção de jogos AAA quanto para produções independentes, como a *CryEngine*, *Frostbite*, *Source Engine*, *Unity* e *Unreal*.

“Diferentes motores, possuem diferentes características e também tem diferentes plataformas como alvo. As funcionalidades básicas que tornam um aglomerado de ferramentas em um motor, tipicamente incluem, um motor de renderização, um motor de física, gerenciador de arquivos, gerenciador de memória, importador de modelos tridimensionais, importador de imagens animadas para jogos em 2 dimensões, suporte para sons, animações 3D e até mesmo linguagem de script para abstração de serviço.”(INÁCIO, 2022)

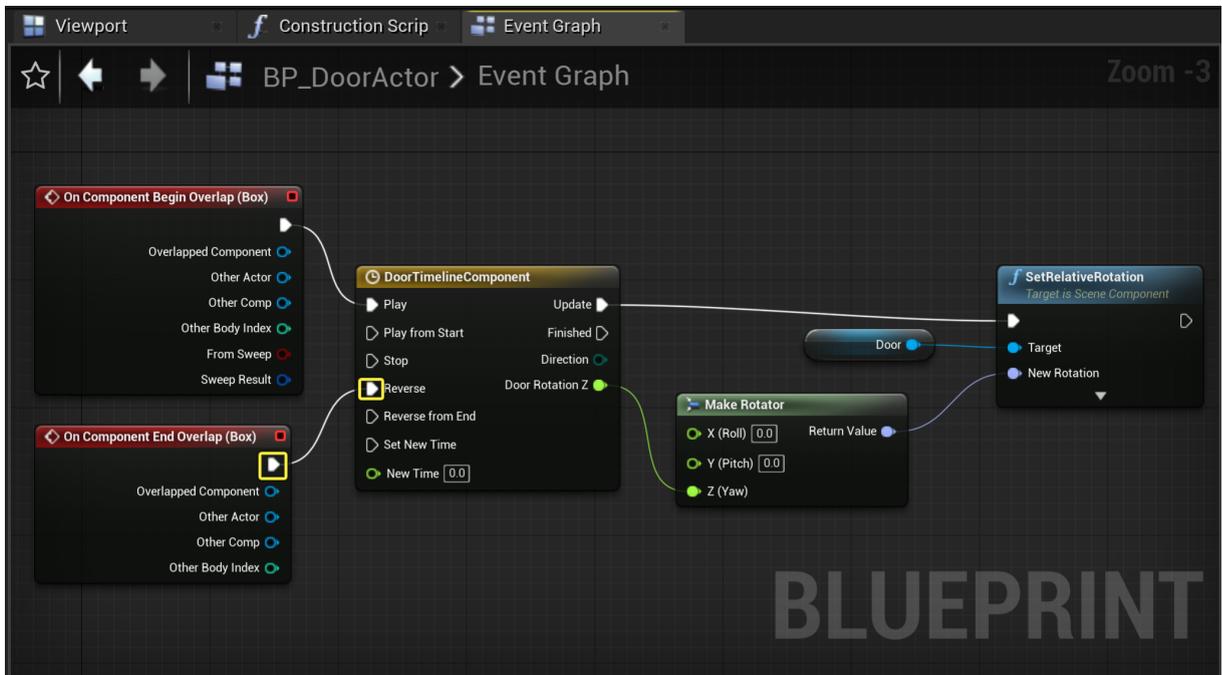
Para esse projeto foi utilizada a *Unreal Engine 5.1*, um motor gráfico disponibilizado gratuitamente pela *Epic Games* e, segundo o próprio site da desenvolvedora:

“A *Unreal Engine* é um conjunto completo de ferramentas de criação para desenvolvimento de jogos, visualização arquitetônica e automotiva, criação de conteúdo linear de cinema e televisão, produção de eventos ao vivo e transmissão, treinamento e simulação e outros aplicativos em tempo real”(EPIC GAMES, 2023).

A *Unreal* utiliza a linguagem de programação C++, mas a *engine* também possui um sistema chamado *Blueprints* (Figura 5), uma interface de *script visual* baseada em *nodes* (DROZINA e OREHOVAČKI. 2018) e que utiliza o método de programação orientada a objeto e possibilita que pessoas que não tenham o domínio da programação possam criar usando a plataforma. O fato de ser um motor

gratuito com um sistema de *script visual* levou à escolha dessa *engine* para o desenvolvimento do protótipo neste projeto.

Figura 5 - Exemplo visual do sistema de *Nodes* de uma *Blueprint*



Fonte:

<https://docs.unrealengine.com/4.27/en-US/ProgrammingAndScripting/Blueprints/UserGuide/Timelines/Examples/OpeningDoors/>

Um jogo normalmente é constituído de elementos tanto visuais quanto sonoros, esses elementos são chamados de *assets*. No caso deste projeto, os *assets* visuais foram compostos majoritariamente de objetos em 3D.

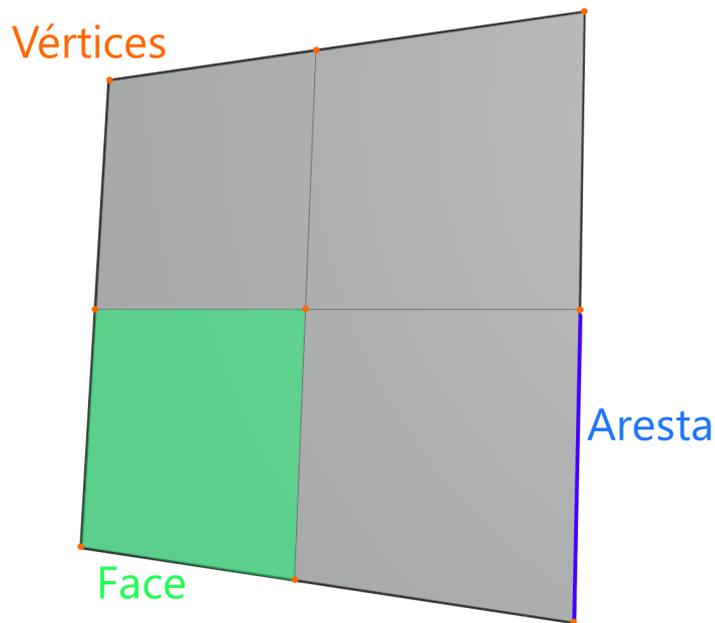
2.3 MODELO 3D

Um modelo 3D é uma representação digital tridimensional de um objeto, criada por meio de um *software* de modelagem 3D. Esses modelos são construídos através de três estruturas básicas: vértices, arestas e faces, que formam a malha, uma superfície que delimita e dá forma ao objeto em um espaço virtual. (BLENDER FOUNDATION, 2023)

O vértice é a parte mais elemental de um modelo, sendo representado por um simples ponto dentro de um espaço virtual, já a aresta é a linha reta formada pela

conexão de dois vértices, e a face é a superfície que se forma ao se conectar no mínimo 3 vértices (BLENDER FOUNDATION, 2023). Na Figura 6 vemos um objeto que possui 9 vértices, 12 arestas e 4 faces.

Figura 6 - Exemplo de um objeto que possui vértices, arestas e faces



Fonte: Autor, 2023.

2.4 MODELAGEM EM *BOX MODELING*

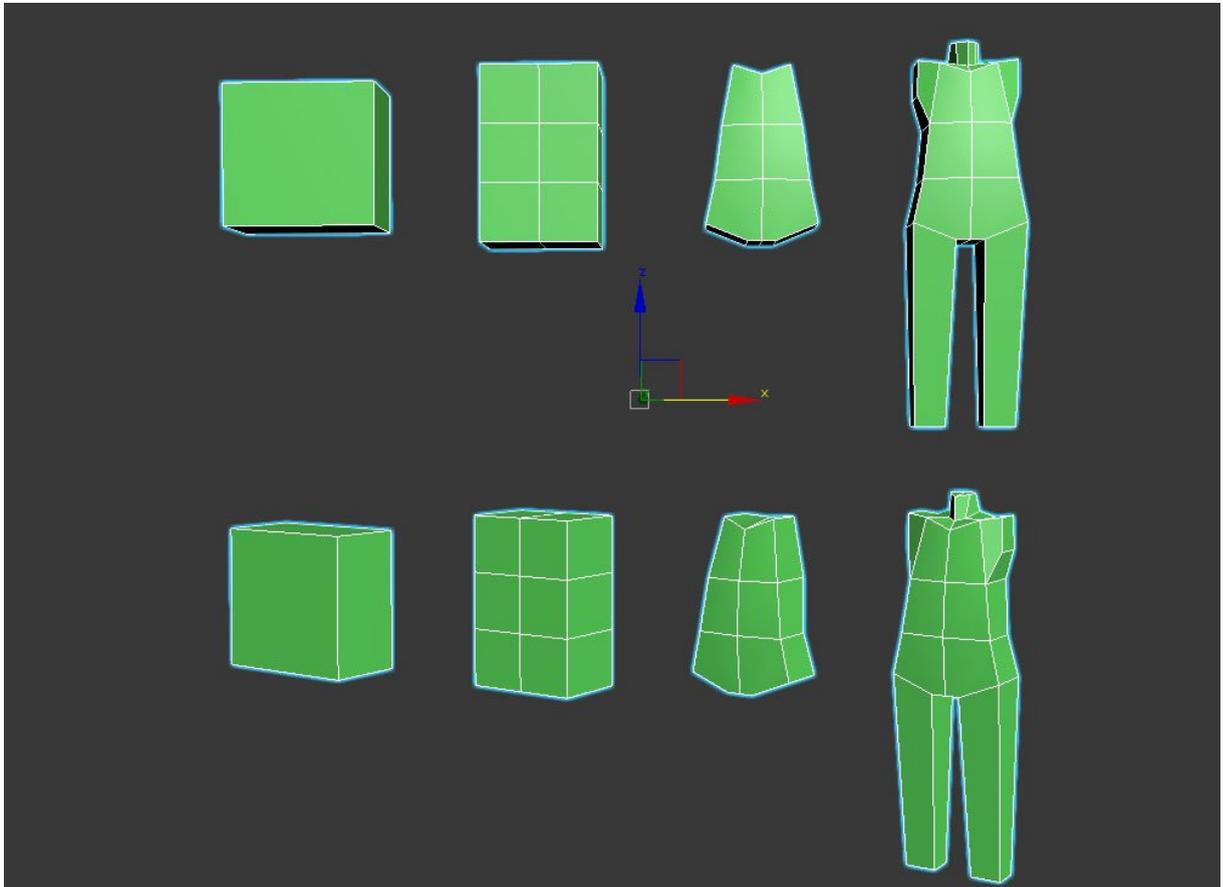
A modelagem é o processo de definir o volume da superfície de uma malha (EMERSON, 2021). Existem diferentes maneiras de fazer a modelagem de um objeto, neste projeto foi utilizado apenas o *box modelling* para tal.

“A modelagem de malhas é iniciada tipicamente usando uma Primitiva de malha como formato de base para a modelagem (por exemplo um círculo, um cubo, um cilindro, etc...). A partir de então, você pode iniciar a edição para a criação de um formato maior e mais complexo” (BLENDER FOUNDATION, 2023).

Dessa forma, *Box Modeling* consiste em trabalhar com uma geometria primitiva inicial e ir gradualmente ajustando a malha, subdividindo, redimensionando, extrudando e utilizando qualquer outro recurso disponível no *software* de

modelagem para adicionar detalhes e obter a forma desejada. É possível observar um exemplo do processo de *Box Modeling* na Figura 7.

Figura 7 - Processo de *Box Modeling*



Fonte: <https://twitter.com/pechetan/status/864487303854432256/photo/1>

2.5 TOPOLOGIA

A topologia em modelos 3D refere-se à organização e estrutura dos polígonos na malha tridimensional. Isso envolve a forma como os polígonos são conectados uns aos outros, o alinhamento das arestas e a distribuição dos vértices na superfície do objeto. Os triângulos são simples e fáceis de calcular, mas possuem uma limitação em termos de deformação mais suaves, já os quadriláteros oferecem maior controle de deformação e são preferidos para animação e modelagem por subdivisão por apresentarem melhores resultados. A escolha entre triângulos e quadriláteros depende do contexto e das necessidades específicas do projeto.

“Triângulos são sempre planos e portanto fáceis de calcular. Por outro lado, quadrados deformam bem e são portanto preferidos para animação e modelagem por subdivisão” (BLENDER FOUNDATION, 2023).

A topologia adequada é importante para obter uma malha suave e sem irregularidades, isso impacta em processos como animação, texturização e otimização de renderização do modelo.

“Existem limites para a manipulação e renderização de vértices em tempo real pela *GPU*. Isto significa que o artista ou designer deve considerar o número de polígonos e vértices ao produzir seu personagens, objetos e terrenos”(NETO, 2017).

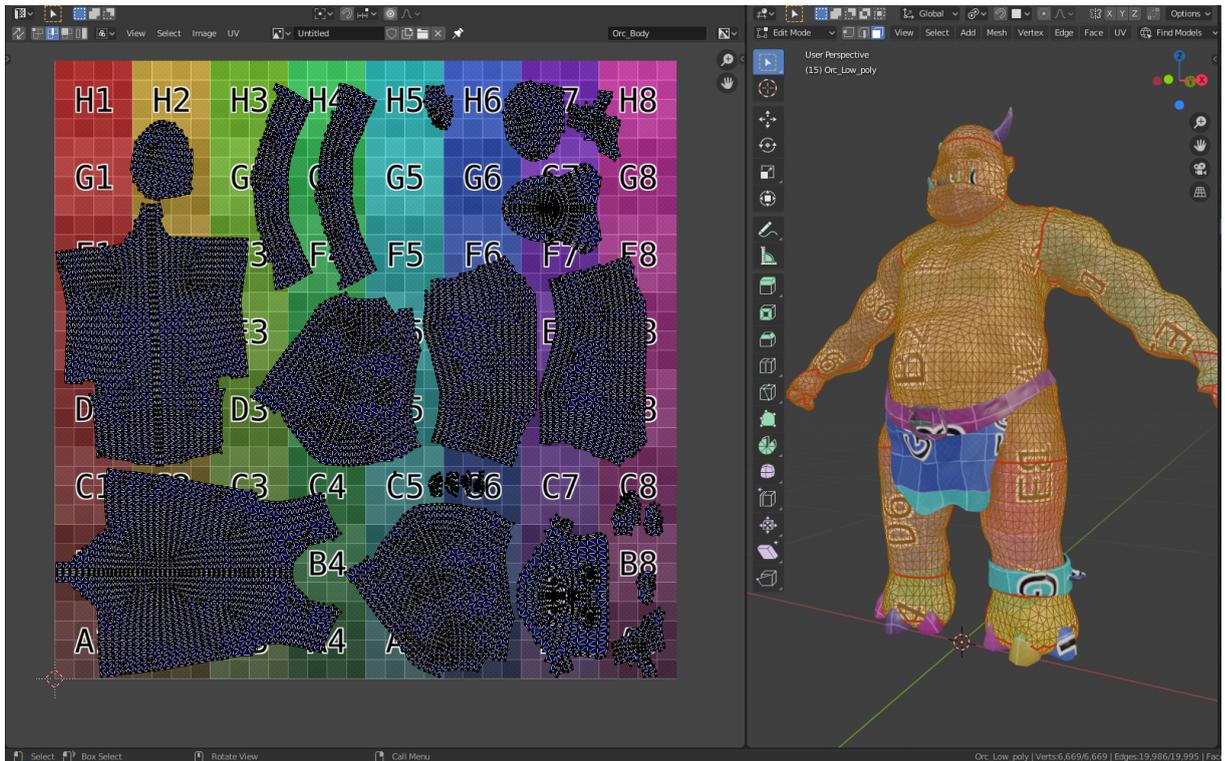
2.6 TEXTURIZAÇÃO

O primeiro passo para a texturização é a realização do processo de abrir a geometria de um objeto 3D em uma representação 2D, o *unwrap*. Ele permite o mapeamento preciso das coordenadas *UV*, que equivalem às coordenadas X,Y de um plano cartesiano, sendo o resultado final dessa abertura o *uv map*.

UV Maps que são mapas bidimensionais que são aplicados sobre um modelo tridimensional e onde cada ponto no mapa 2D corresponde a um ponto na malha 3D, possibilitando que uma imagem bidimensional possa ser utilizada como textura do objeto tridimensional. (JAMES, 2012)

A Figura 8 ilustra o conceito tendo à direita um modelo 3d, e à esquerda a representação do *uv map* com uma textura contendo diversas cores e caracteres. A área correspondente de cada uma dessas coordenadas da imagem é projetada diretamente no modelo.

Figura 8 - Exemplo de como o *Uv Map* e a malha funcionam na textura



Fonte: <https://community.gamedev.tv/t/packed-uv-map-orc/134566>

2.7 ANIMAÇÃO 3D

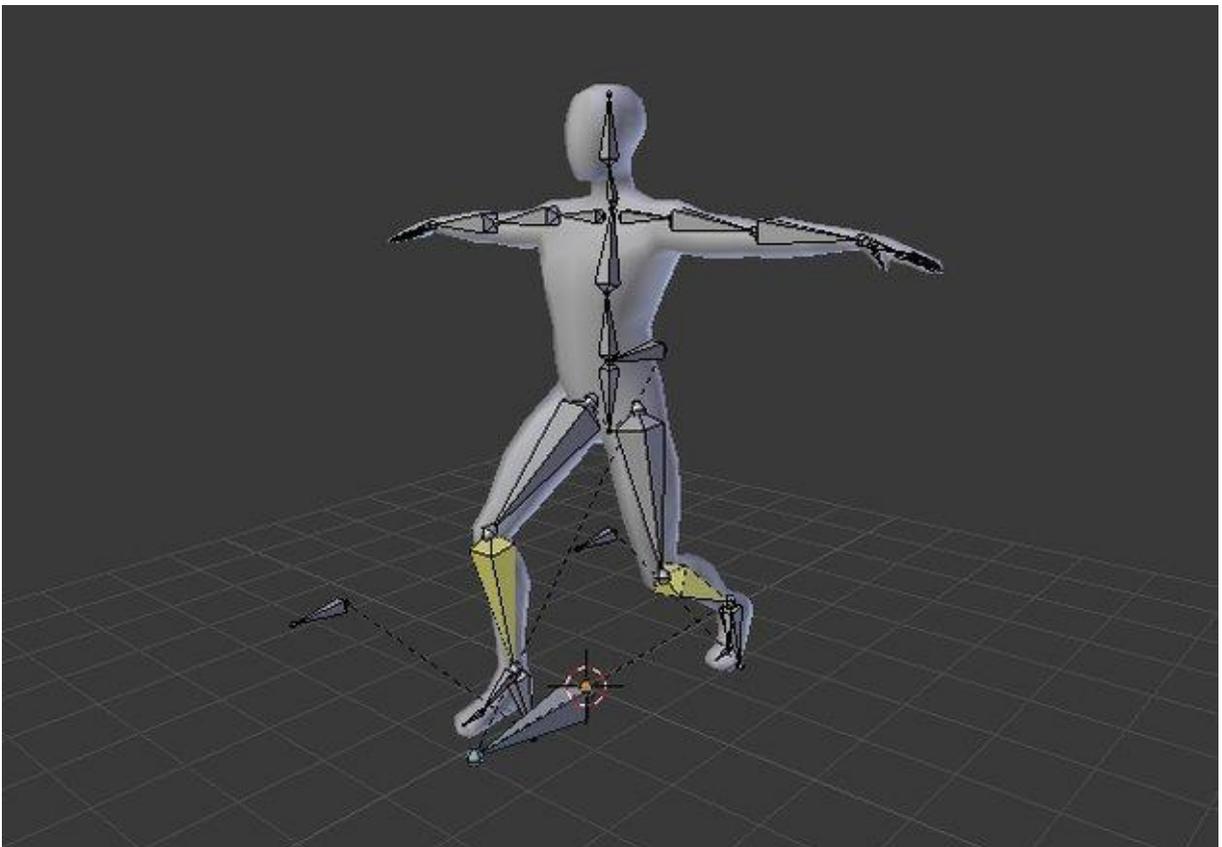
A animação 3D é uma técnica que consiste em posicionar o personagem/objeto em diversas poses estáticas que, quando exibidas em sequência, criam a sensação de movimento. Cada pose é alocada em um *keyframe* (quadro-chave) que “é simplesmente um marcador de tempo que guarda o valor de uma propriedade.” (BLENDER FOUNDATION, 2023)

Principalmente para a animação de personagens, é essencial criar um sistema que permita a manipulação da malha de forma orgânica. Para isso, é utilizado a *armature*, ou *Rig*, um esqueleto de animação composto por *bones* (ossos) que são peças posicionadas dentro da malha e que criam essa estrutura móvel do personagem. Um *Rig*, é uma estrutura hierárquica, semelhante à uma árvore. Em um personagem humanóide (Figura 9), ele tende a se adequar à estrutura óssea real, com exceções dependendo da situação (VENUTA, 2014).

Caso seja necessário, ainda existe a opção de adicionar o *IK* (Inverse Kinematics), uma função que permite controlar os movimentos de forma mais

intuitiva por meio do posicionamento de um ponto final desejado, enquanto o restante da cadeia de ossos é ajustado automaticamente para alcançar essa posição. Diferentemente do *FK (Forward Kinematics)*, onde cada osso precisa ser rotacionado individualmente para criar o movimento desejado. O *IK* permite controlar a extremidade do membro de forma que possa ser fixada em posições estratégicas, favorecendo animações que seriam muito difíceis de produzir em *FK*, como a fixação dos pés do personagem no chão durante uma caminhada por exemplo.

Figura 9 - Exemplo de um Rig



Fonte:

<https://asuevents.asu.edu/content/fablab-share-armor-bone-structure-animation>

Com os ossos posicionados no modelo, é necessário definir o quanto cada um deles irá influenciar no movimento da malha, definindo valores a cada vértice para isso. Essa etapa é conhecida como *weight paint*. Definir o peso de cada um serve para que, ao mexer os *bones*, a malha se deforme de maneira coesa.

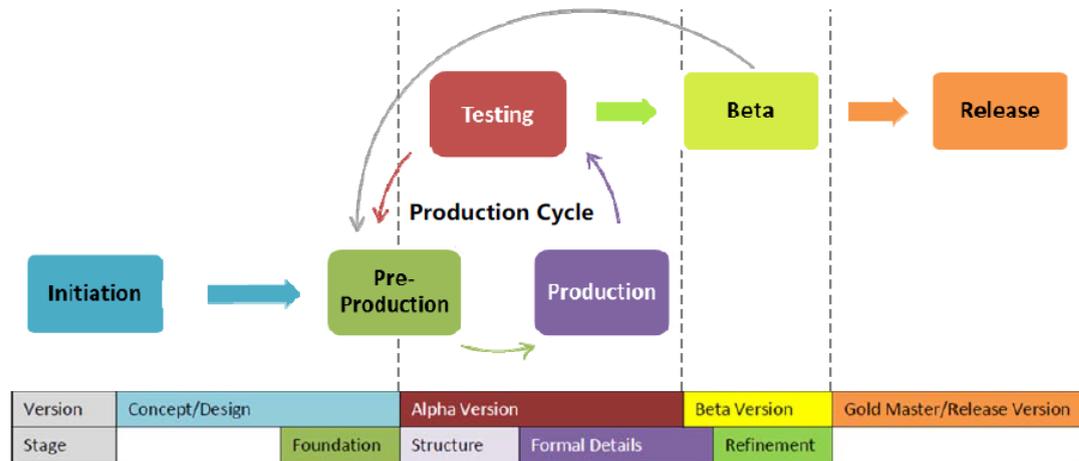
3 DESENVOLVIMENTO

Para a produção do jogo em questão foram utilizados como base 2 pipelines: o de Wardrip-Fruin e Montfort (2003), e o de Chen (2009). No intento de desenvolver um *Serious Game* que possibilite a exploração de um ambiente de uma maneira mais lúdica, o presente projeto adaptou a metodologia de desenvolvimento de jogos de Wardrip-Fruin e Montfort (2003), que é baseada em cinco fases: pré-produção, produção, testes, avaliação e manutenção. A partir disso, propõe-se a utilização dos conceitos metodológicos apresentados por Michael e Chen (2009), adaptando os processos de desenvolvimento de jogos aos *Serious Games* com um objetivo instrucional. Dessa forma, as etapas foram:

- Definição dos objetivos educacionais: estabelecer os objetivos educacionais do jogo, como a criação de uma visita gamificada que tem como objetivo permitir mais um canal de comunicação entre universidade e sociedade .
- Levantamento de requisitos: coletar os requisitos do jogo, como os locais, salas e personagens a serem apresentados no projeto.
- Desenvolvimento do design: definir a interface gráfica, a navegação do jogador e a dinâmica do jogo.
- Desenvolvimento do conteúdo: elaborar o conteúdo do jogo, incluindo os textos, imagens, objetos tridimensionais e animações.
- Desenvolvimento do software: definir linguagens de programação e ferramentas de desenvolvimento específicas.
- Testes e validação: garantir que o jogo atenda aos objetivos educacionais e aos requisitos estabelecidos por meio de testes.

A metodologia de Ramadan e Widiyany (2013) que propõe um modelo de produção (Figura 10), permitiu que o autor produzisse o jogo de acordo com as suas necessidades e com a possibilidade de mudanças, evitando que a falta de conhecimento em alguns *softwares* pudessem atrapalhar muito o andamento do protótipo.

Figura 10 - Exemplo do modelo de produção



Fonte: Ramadan e Widiny, 2013

A. Iniciação

O processo inicial de criação de um jogo envolve o desenvolvimento de um conceito básico que define o tipo de jogo a ser criado. Nesta etapa inicial, estabelece-se o conceito do jogo e realiza-se uma descrição simples do mesmo.

B. Pré-produção

A pré-produção é uma fase inicial no ciclo de produção de um jogo. Durante essa fase, ocorre a criação do design do jogo e a elaboração de um protótipo inicial.

C. Produção

A produção é o processo que envolve a criação de *assets*, desenvolvimento da programação e integração desses elementos. Aqui o protótipo criado anteriormente é refinado.

D. Testes

Os testes, referem-se à avaliação interna de jogabilidade realizada pelos próprios desenvolvedores.

E. Beta

A fase *Beta* é realizada para conduzir teste do jogo com terceiros.

F. Lançamento

O lançamento é o momento em que a versão final do jogo foi concluída e está pronta para ser disponibilizada ao público.

3.1 PRÉ-PRODUÇÃO

O primeiro passo na criação do projeto foi decidir o que seria feito e definir os objetivos a serem alcançados para desenvolver um jogo em 3D e que tivesse o campus da Universidade Federal de Santa Catarina (UFSC) como cenário principal.

Para organizar as ideias, foi feito o *GDD*, definindo assim a elaboração de um jogo simples que não exigisse habilidades em videogames para jogar, tornando-o mais acessível, como uma espécie de tour gamificado, em que o personagem principal fosse um Ufscão. Ufscão é uma maneira que as pessoas que frequentam o campus se referem aos cachorros que andam/moram pela universidade. As referências iniciais demonstradas na Figura 11, foram alguns jogos que possuem animais como protagonistas, entre esses jogos estão *Stray* (2022), *Untitled Goose Game* (2019) e *Ōkami* (2006).

Figura 11 - Referências de jogos



Fonte: Autor, 2023.

A ideia foi produzir um jogo na perspectiva de um cachorro, cuja percepção demonstrasse a grandeza da universidade. O personagem poderia caminhar pelo cenário e interagir com alguns objetos e outros personagens. O cenário do jogo será dividido em níveis. O maior nível contém uma grande zona envolvendo áreas externas do campus. As áreas internas de cada prédio serão contidas em níveis menores, sendo carregados quando o jogador entrar no respectivo prédio. Foi delimitado um recorte envolvendo um escopo de produção exequível por meio de um protótipo, contendo um prédio e algumas salas de maior relevância para o curso de Animação.

Dentro da parte gráfica foi realizada uma pesquisa de possíveis estilos gráficos a serem utilizados e, dentre as diversas possibilidades, foram escolhidos alguns para guiar o projeto. Parte da ideia do jogo é que ele fosse acessível para o maior número de pessoas e funcionasse bem em uma boa quantidade de aparelhos. Para isso, foi importante se atentar se o jogo era leve, exigindo menos do processador e da placa gráfica. Com isso em mente, foi definido que o cenário seria o mais *low poly* possível, liberando assim uma maior quantidade de polígonos para os personagens.

Outro ponto importante foi o visual ser lúdico, e para isso foram buscadas inspirações em jogos mais estilizados. Para alcançar esse efeito foi decidido que, tanto o cenário quanto os personagens, teriam as texturas não metálicas com o *roughness* alto, ficando menos reflexivas. Já para os personagens, seria utilizado um efeito chamado *Cell Shading*, (Figura 12), também chamado de *toon shading* ou *cel/toon shader*, é uma técnica 3D baseada em um método específico de sombreamento. Seu objetivo é recriar a aparência das animações 2D tradicionais, utilizando cores planas para sombrear objetos 3D de maneira não realista. Essa técnica é amplamente reconhecida como um estilo artístico que confere aos gráficos 3D uma aparência cartunesca, empregando texturas coloridas distintas e contornos para simular linhas de desenho. (LUQUE, 2012)

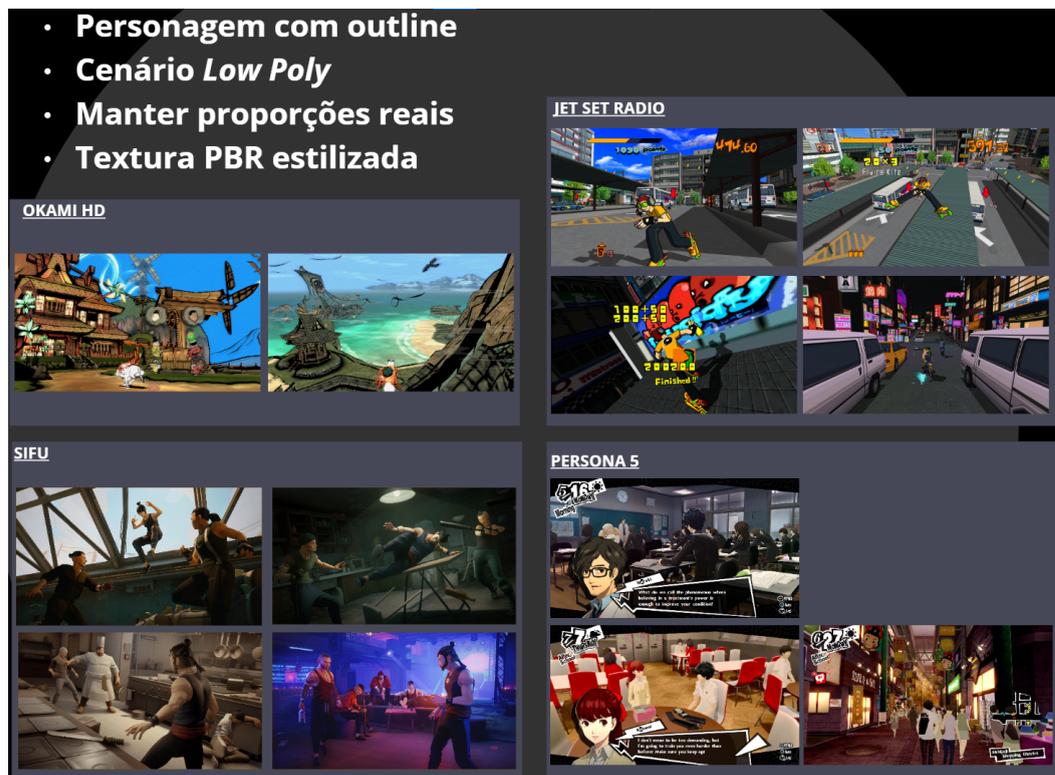
Figura 12 - Exemplo de personagens com *Cell Shading*



Fonte: Genshin impact, Hoyoverse, 2020

Foram então reunidos alguns jogos, (Figura 13), que possuem um ou mais desses elementos para serem utilizados de referências iniciais.

Figura 13 - Painel de referência visual



Fonte: Autor, 2023.

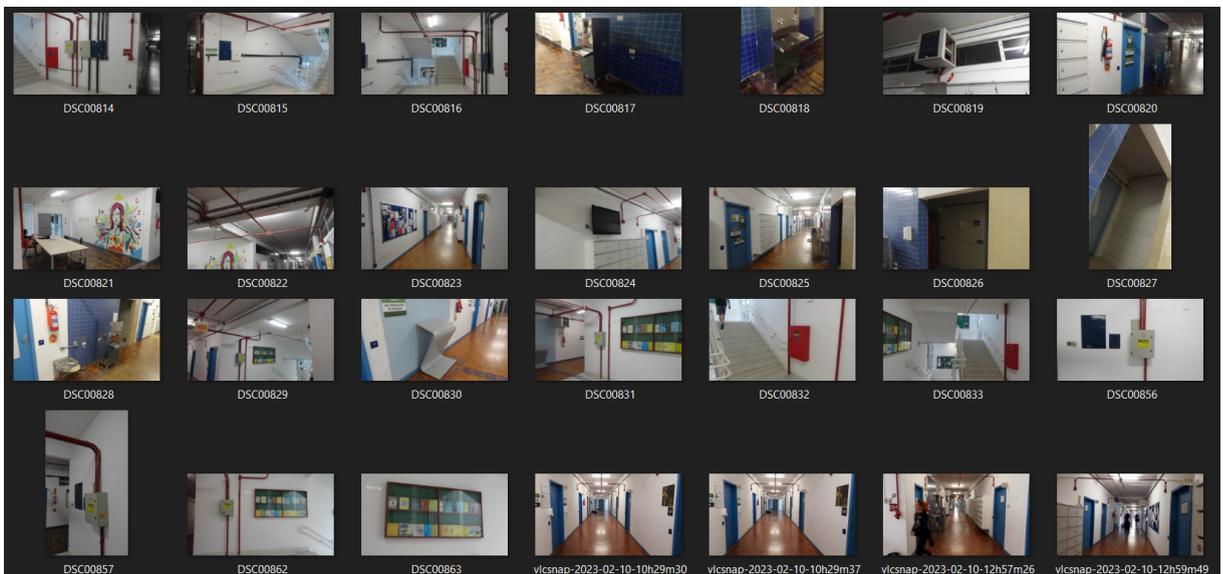
3.2 PRODUÇÃO

A etapa de produção envolveu a produção dos assets, o processo começou com a blocagem, criando representações básicas dos assets usando formas geométricas simples para definir a estrutura geral do objeto ou cena, seguindo pela etapa de refinamento texturização até a implementação desses modelos na engine.

3.2.1 BLOCAGEM DO CENÁRIO

O cenário escolhido dentro do escopo foi o Bloco A do CCE (Centro de Comunicação e Expressão), que foi preferido devido à familiaridade do autor com o local. Inicialmente, o jogo teria o térreo, os 2 andares do prédio, e algumas salas. A primeira etapa na construção do cenário foi a utilização de referências. Como a ideia era reproduzir da maneira mais fiel possível um local real, necessitou-se de fotos (Figura 14) e medidas dos locais a serem modelados para essa etapa da pré-produção.

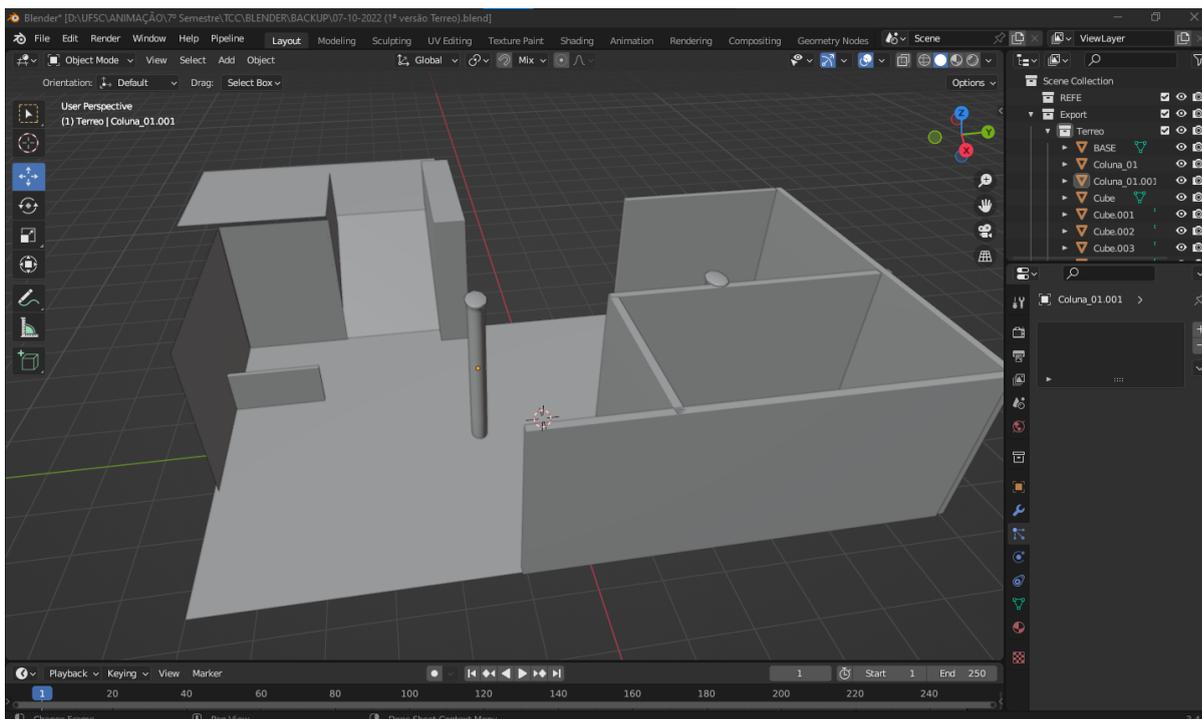
Figura 14 - Fotos de referência da universidade



Fonte: Autor, 2023.

Com as medidas e referências disponíveis, iniciou-se a segunda etapa, a blocagem. Utilizando o *Blender* (Figura 15), foram adicionadas formas primitivas para definir as proporções das estruturas dentro do ambiente 3d.

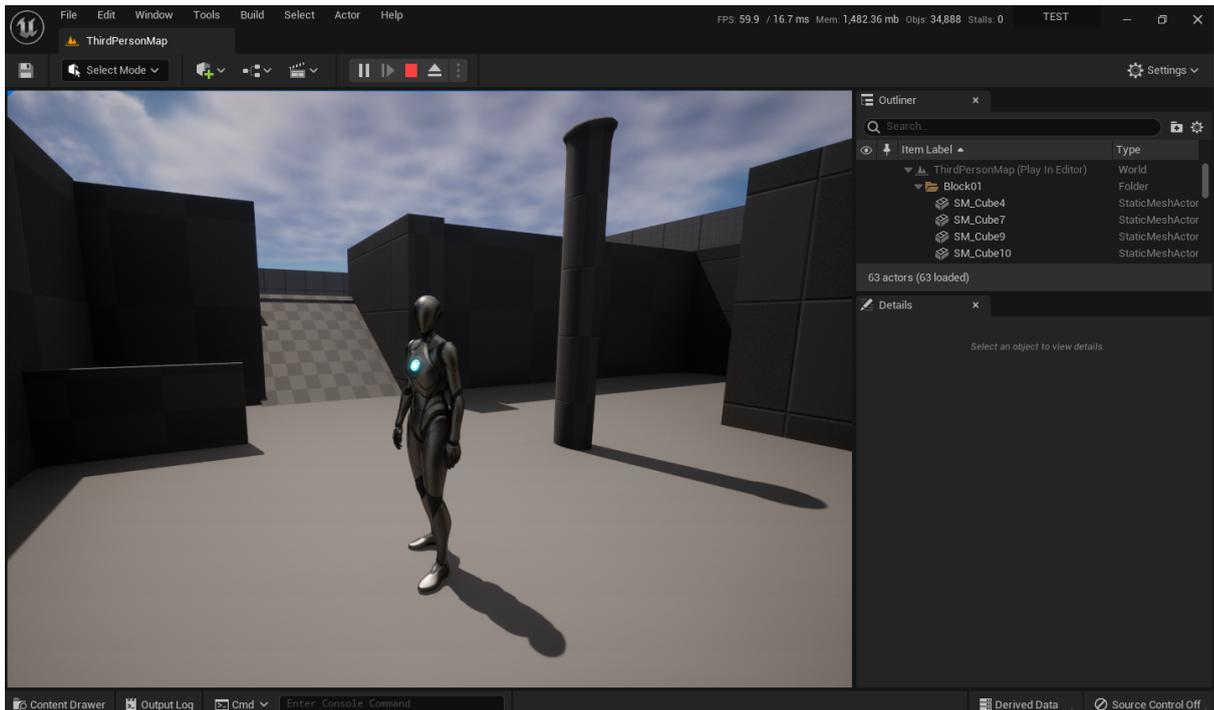
Figura 15 - Etapa de blocagem do cenário no software Blender



Fonte: Autor, 2023.

Corredores, escadas e salas do térreo foram adicionados nesta versão do arquivo. Nessa etapa foram feitos os primeiros testes de importação para a *UE5* (Figura 16), além da exploração de como o programa funciona. Foi observado que a utilização das formas primitivas de planos não era a mais eficiente nessa etapa, pois esses objetos na *unreal* não tinham física de colisão, uma função essencial, já que ela permite que dois objetos não ocupem o mesmo lugar no espaço tridimensional do jogo. Para contornar isso e manter a quantidade de polígonos a mais baixa possível, as peças do ambiente começaram a ser modeladas como cubos e, em seguida, deletou-se as faces que não seriam vistas pelo jogador, permitindo a importação desses objetos com a física correta na *engine*.

Figura 16 - Primeiro teste de importação do cenário para a Unreal



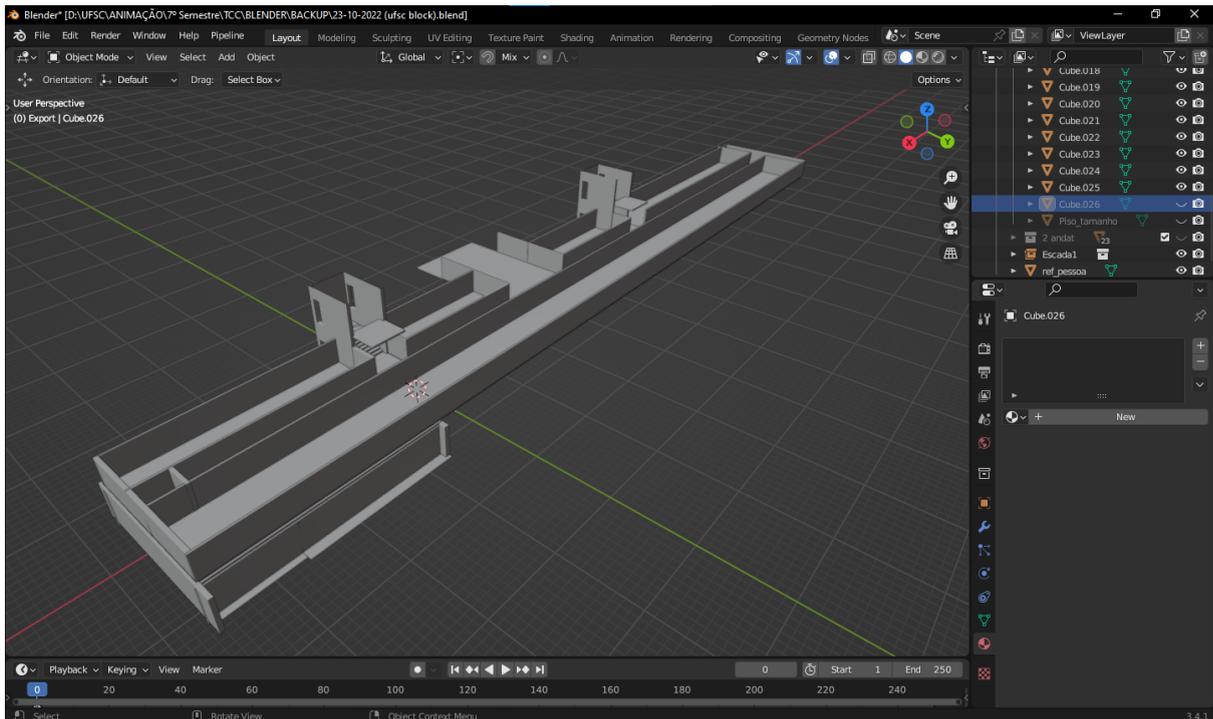
Fonte: Autor, 2023.

Devido ao *addon*² chamado *send2ue*, foi possível utilizar um *workflow* que permitiu a modificação dos objetos no *Blender* e a substituição deles na *engine* de maneira mais simples, desde que não fossem renomeados, sem a necessidade de salvar o arquivo e importar manualmente para a *Unreal*. Isso permitiu que ajustes fossem sendo feitos no *Blender*, importando para *Unreal* para verificar como ficava em jogo, e reajustando novamente no *Blender* quando necessário.

Com a adição de mais elementos, a cena começou a ficar grande (figura 17), e para facilitar nas etapas seguintes, iniciou-se a renomeação dos arquivos colocando um prefixo para cada tipo de objeto.

² *Addons* são complementos ou funcionalidades extras que podem ser instalados/ativados em um aplicativo.

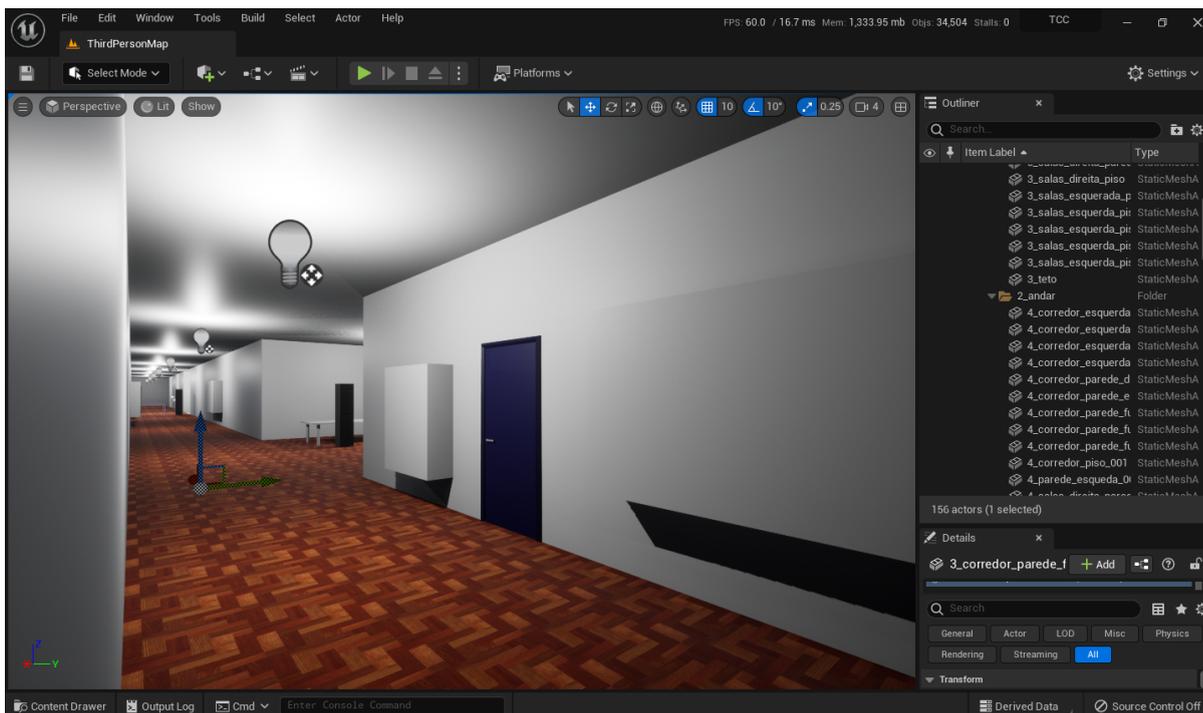
Figura 17 - Blocagem do primeiro andar do prédio



Fonte: Autor, 2023.

Com a estrutura do prédio já finalizada, iniciou-se os ajustes da etapa de blocagem dos objetos demonstrados na Figura 18, como mesas, portas, armários e luminárias. Cada objeto foi modelado em um arquivo separado no *Blender* e exportado para a *Unreal* da mesma maneira que a estrutura do prédio.

Figura 18 - Blocagem inicial dos objetos no cenário

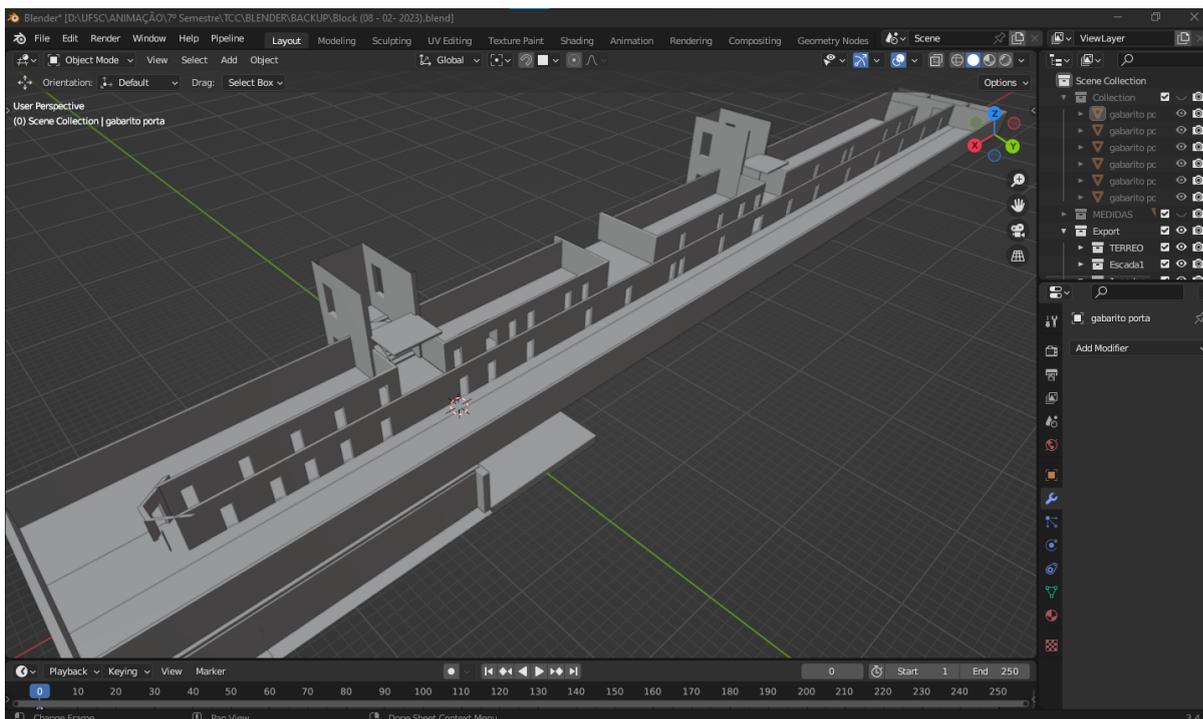


Fonte: Autor, 2023.

Para a adição dos buracos para a porta, foi utilizado o modificador *boolean*³. Apesar da forma do objeto ter ficado visualmente correta, ao abrir o menu de edição para adicionar texturas, foi constatado que a topologia do objeto estava distorcida e apresentava problemas para a adição de texturas. Foi necessário refazer todas as malhas novamente com os buracos já modelados, e o resultado pode ser visto na Figura 19.

³ O modificador boolean permite realizar adição, subtração ou a intersecção de um objeto em outro.

Figura 19 - Processo de blocagem no Blender

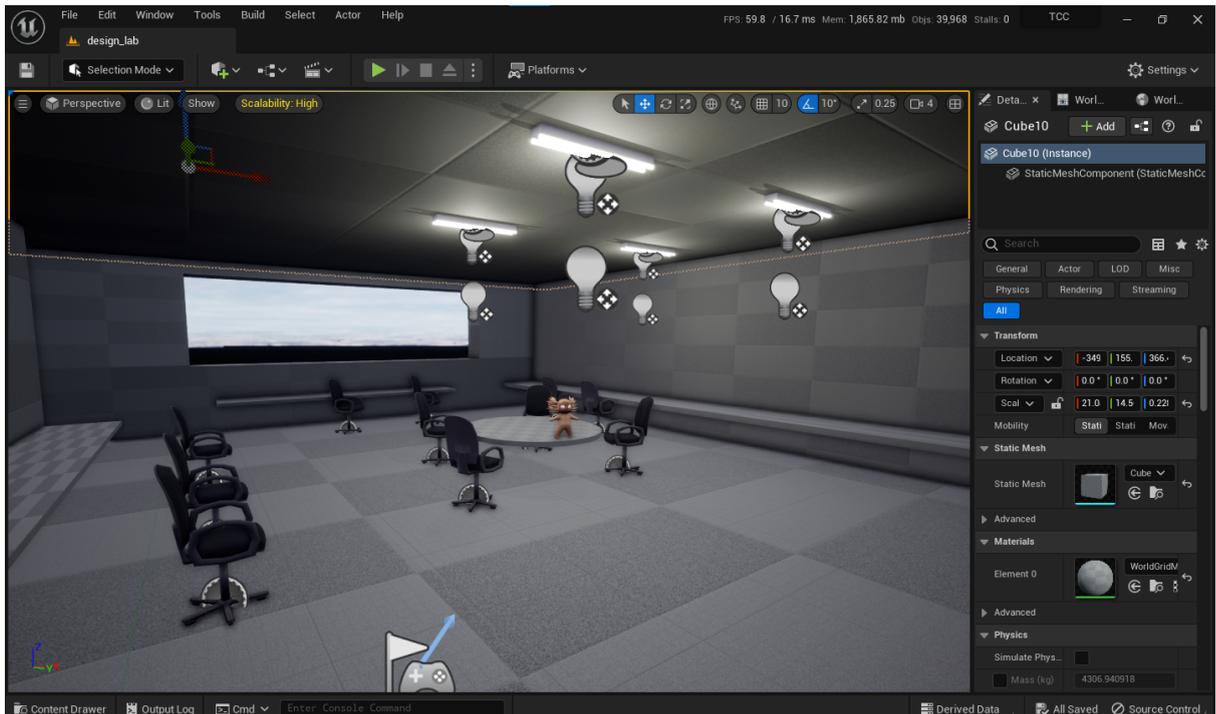


Fonte: Autor, 2023.

Usando o *Smart UV Project*, a opção automática de *unwrap* do *Blender*, foram montados os *UV Maps* das estruturas feitas até então para permitir a texturização na própria engine.

A modelagem das salas seguiu um caminho diferente da estrutura principal, o primeiro passo foi a criação de um nível novo que foi nomeado como *design_lab*. Com um cenário vazio, iniciou-se a blocagem diretamente na *Unreal* por meio dos primitivos da própria engine. Em seguida, foram adicionados alguns *props* apenas para fins de visualização. A primeira versão do cenário (Figura 20) foi feita como uma cópia de baixa fidelidade do local real apenas para adicionar o sistema de troca de níveis.

Figura 20 - Blocagem cenário Design Lab



Fonte: Autor, 2023.

Com base nas referências fotográficas e medidas, foram feitas alterações no espaço para representar melhor o local. O teto e a distância entre as paredes foram reduzidos, o tamanho da janela foi modificado e foram adicionados os pilares estruturais da sala. Além disso, foram aplicadas as texturas nas superfícies do piso e paredes, visando reproduzir de forma fiel a aparência desses elementos. O resultado dessa etapa pode ser visto na Figura 21.

Figura 21 - Design Lab



Fonte: Autor, 2023.

3.2.2 CICLO DE PRODUÇÃO DOS ASSETS

A modelagem dos *assets* seguiu o mesmo ciclo de produção para todos os objetos, dessa forma, a descrição neste artigo se refere às etapas dos objetos de modo geral.

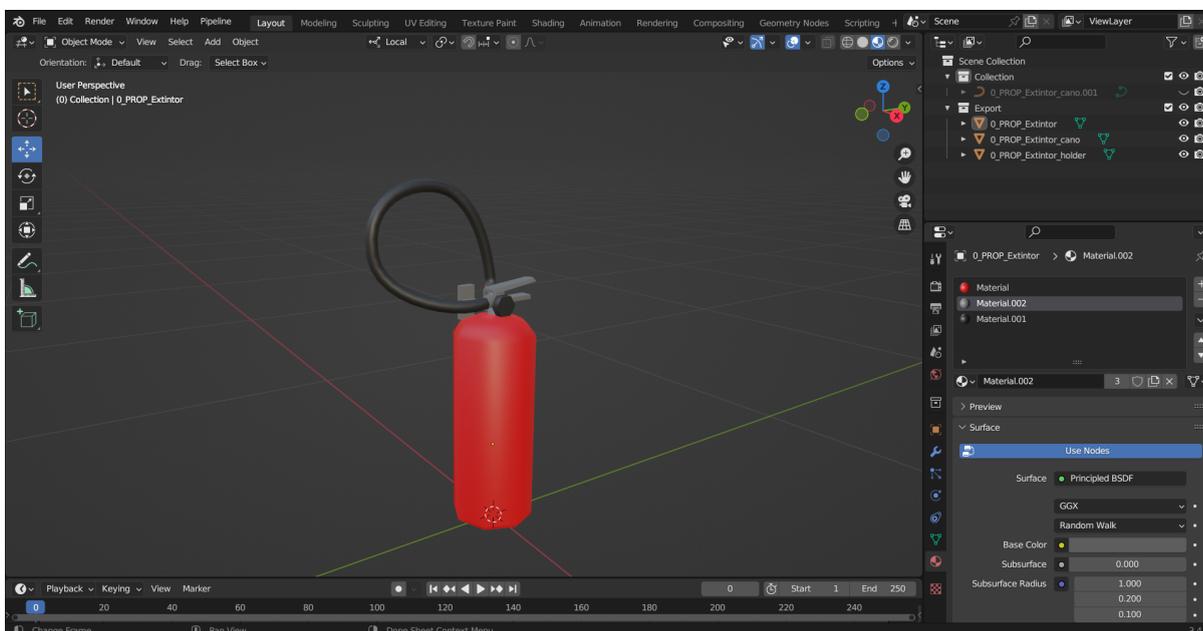
3.2.2.1 MODELAGEM DOS ASSETS

Inicialmente, selecionou-se o objeto a ser modelado e suas respectivas referências visuais por meio de fotos tiradas na universidade, além de imagens buscadas na internet.

Seguindo a proposta gráfica durante a modelagem, buscou-se atingir o objetivo de utilizar o menor número de polígonos possível, mas ainda assim mantendo a forma desejada. Durante o processo de modelagem era escolhida a forma primitiva para editar e dar forma ao objeto, porém, em alguns casos em que o objeto era composto por várias peças, foi utilizada uma primitiva para cada uma dessas partes.

Cada parte do objeto recebeu um material no próprio *Blender* com o intuito de identificar no modelo os diferentes locais para a aplicação de cada textura. Um exemplo do processo da modelagem de um objeto realizada no *Blender* pode ser observado na Figura 22.

Figura 22 - Modelagem de asset



Fonte: Autor, 2023.

Por serem texturizados na *Unreal*, os *UV Maps* foram feitos utilizando a função *Smart UV Project*. Conforme citado anteriormente, nessa etapa também foi

feita a importação para a *Unreal* utilizando o *addon send2ue* para fazer os ajustes no *Blender* quando necessários.

Na *Unreal* foram criadas *blueprints* para cada objeto, o que permitiu a criação de objetos com diferentes funcionalidades dentro da engine com a vantagem de, ao alterar ou adicionar alguma funcionalidade nesse novo objeto criado, ele adicionaria essa modificação a todos os outros.

3.2.2.2 MATERIAIS E TEXTURA DOS ASSETS

A textura dos *assets* foi resolvida de maneiras diferentes dependendo do objeto a ser texturizado, enquanto os personagens receberam pintura a mão, criando um *uv map* manualmente e sendo pintada no *software* Krita.

No cenário e objetos foi usada a opção de *uv map* automática do *Blender*, a *Smart uv unwrap*, e foram texturizadas utilizando o sistema de materiais da *Unreal* por meio de uma sólida como cor base, e adicionando variações na coloração da parede com *noise* texturas. Uma das texturas presentes nas paredes do cenário (Figura 23) foi retirada a partir de uma fotografia do local real.

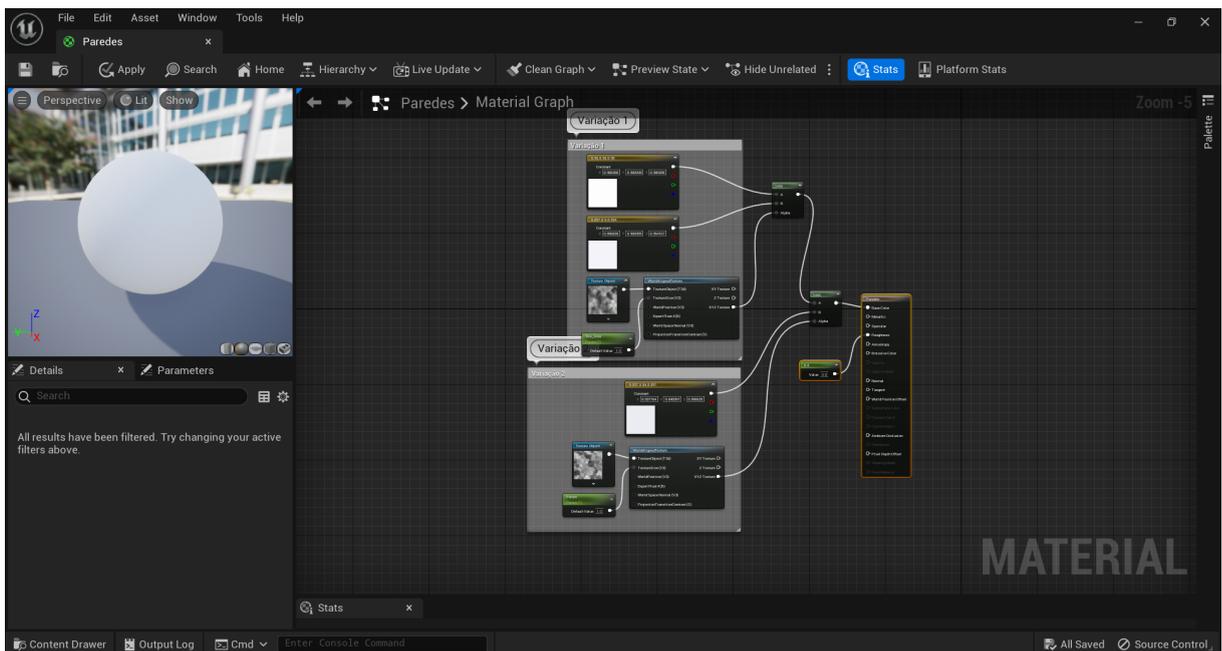
Figura 23 - Textura do mural presente em um dos corredores



Fonte: Autor, 2023.

Para manter a mesma proporção de detalhamento entre os diferentes objetos, foi utilizado o *node* de *World Aligned Texture*, que permitiu ao objeto ignorar o *uv map* existente no modelo, e passou a usar o próprio mundo para o definir o tamanho das texturas. Para estilizar um pouco o cenário, o *roughness* da cena foi aumentado para que os objetos ficassem com um aspecto fosco. O exemplo de como ficou o painel de materiais da parede pode ser observado na Figura 24.

Figura 24 - Painel de materiais da parede na Unreal



Fonte: Autor, 2023.

3.2.3 PRODUÇÃO DE PERSONAGENS

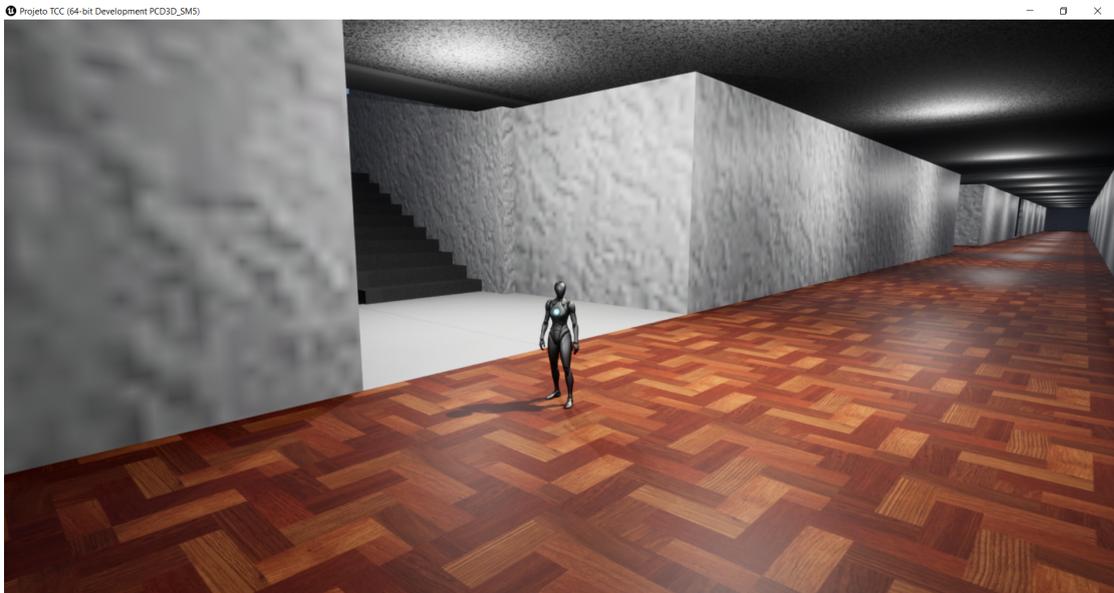
Para esse protótipo foi definido a prototipagem de 2 personagens, um cachorro e o mascote do curso da animação, para produção de ambos seguiu-se um processo similar explicado adiante.

3.2.3.1 UFSCÃO

O Ufscão foi o elemento do projeto que sofreu alterações mais significativas durante a fase de desenvolvimento do protótipo. Inicialmente, o personagem utilizava o personagem padrão da Unreal Engine (figura 25), que foi redimensionado

para proporcionar uma representação aproximada da perspectiva visual do personagem durante a etapa de blocagem.

Figura 25 - Primeira versão jogável do protótipo



Fonte: Autor, 2023.

Para a segunda versão do personagem, foi feita uma modelagem preliminar de teste, ou seja, não houve preocupação com elementos como topologia, texturas e *rigging* (Figura 26).

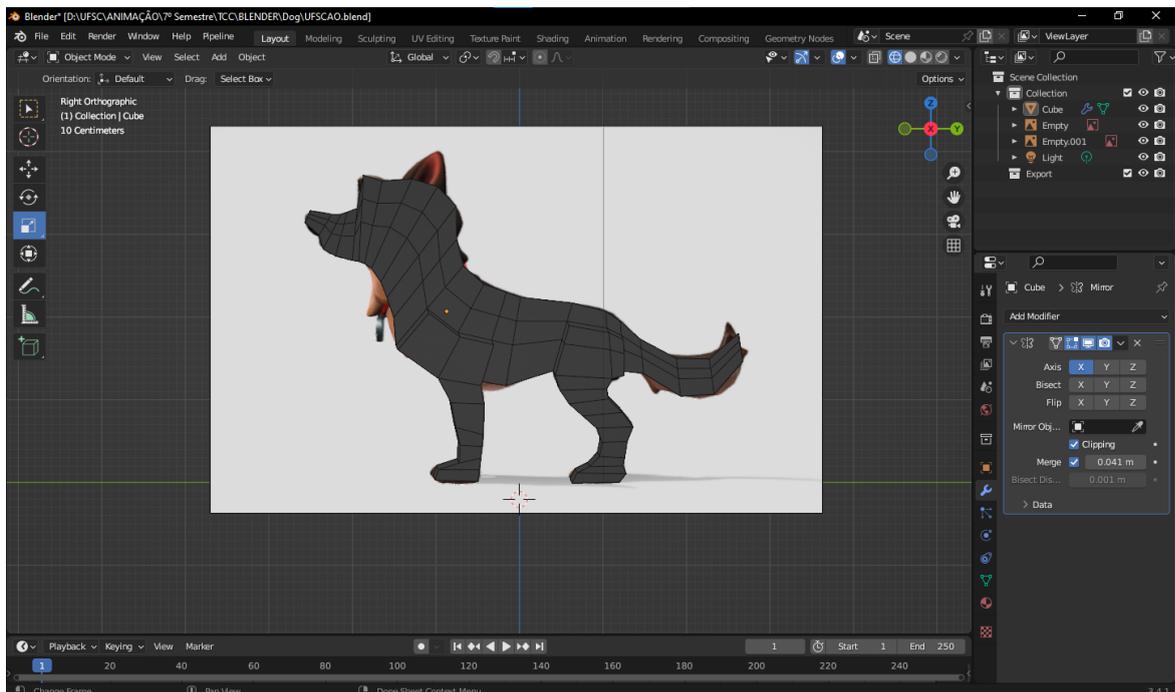
Figura 26 - Segunda versão jogável do protótipo



Fonte: Autor, 2023.

Para a terceira versão, foram utilizadas referências de personagens estilizados como base para a produção do modelo (Figura 27). A técnica de modelagem utilizada foi o *Box Modelling*. No entanto, o modelo ficou com uma aparência incompatível com o que foi buscado para esse personagem, e foi descartado sem seguir para as etapas seguintes da produção.

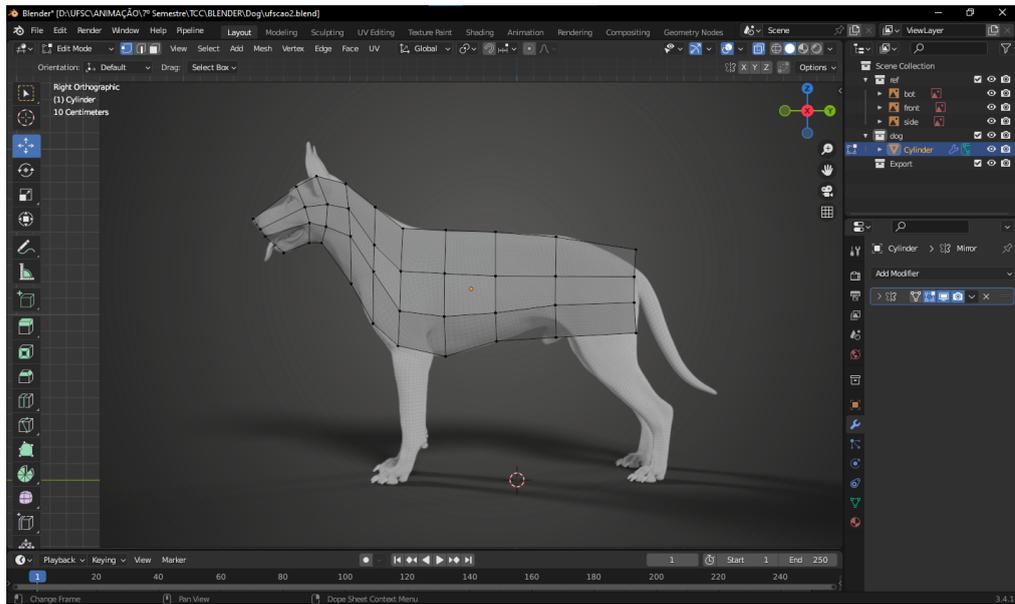
Figura 27 - Modelagem da terceira versão do personagem



Fonte: Autor, 2023.

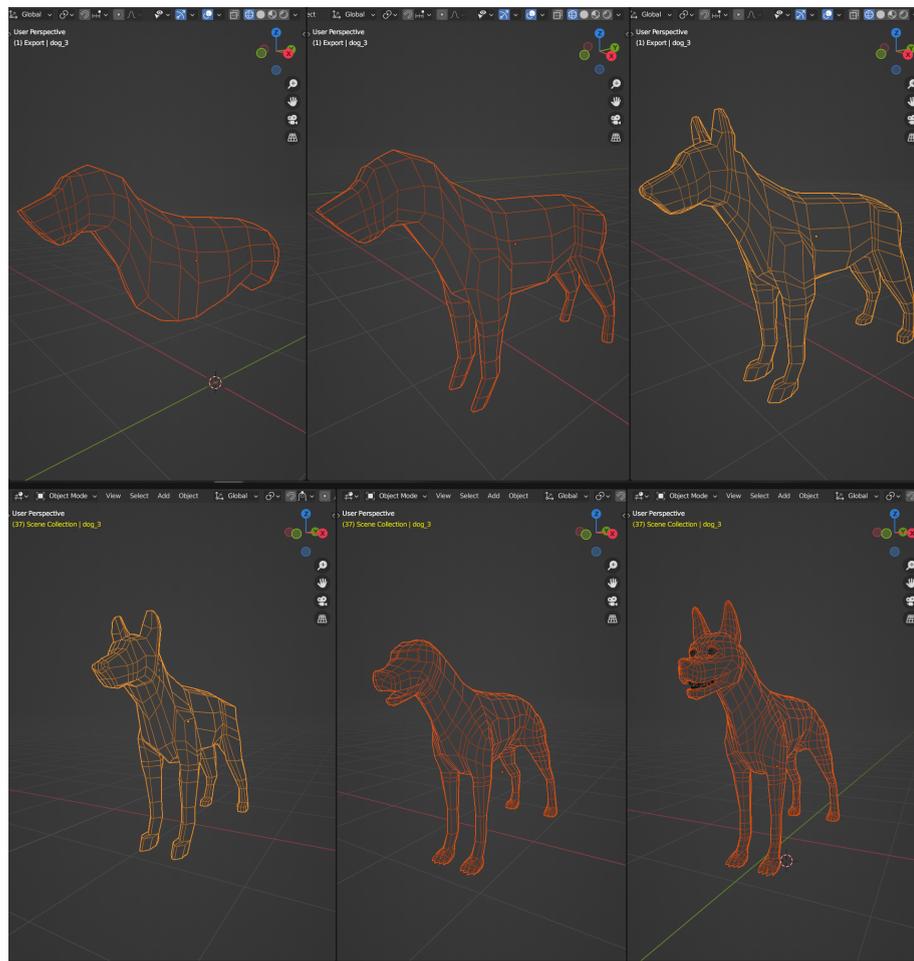
A quarta versão do Ufscão foi concebida utilizando referências mais realistas. Ainda seguindo o processo de *Box Modeling* (Figura 28), a primeira etapa foi a blocagem do personagem com uma quantidade baixa de polígonos para definir a silhueta principal e, aos poucos, foi-se adicionando detalhes. O processo da modelagem pode ser observado em mais detalhes na Figura 29.

Figura 28 - Início da modelagem da quarta versão do personagem



Fonte: Autor, 2023.

Figura 29 - Processo de modelagem de personagem

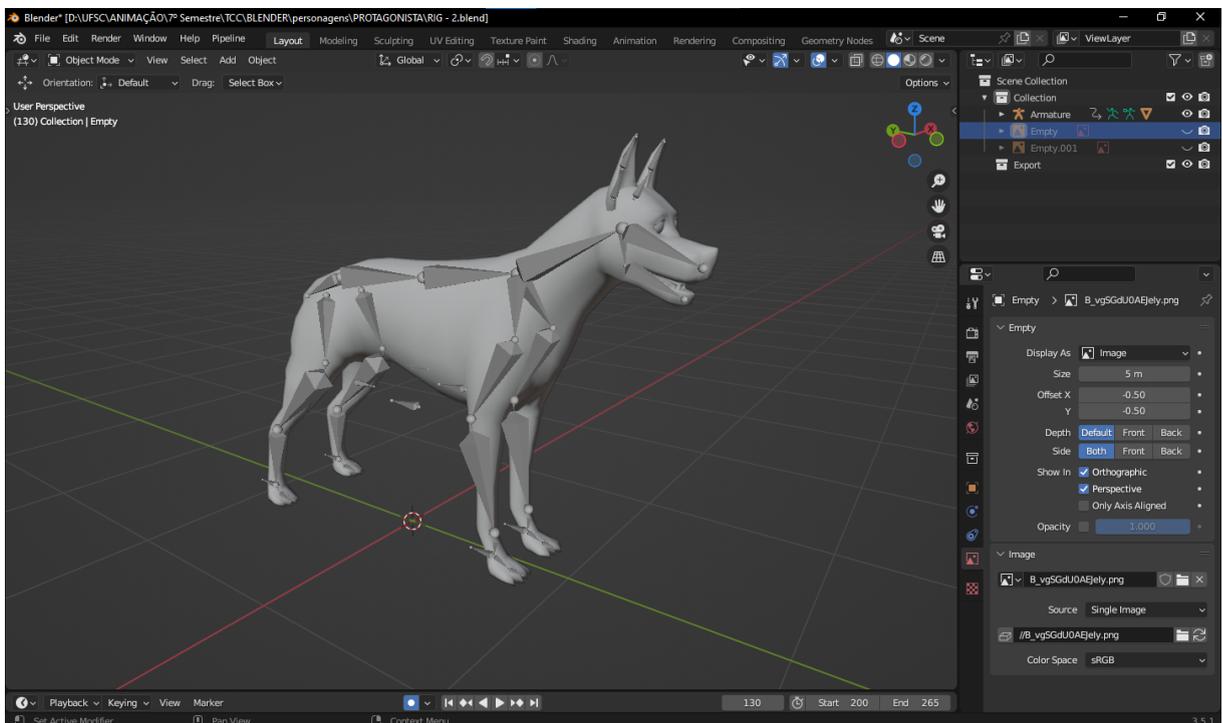


Fonte: Autor, 2023.

Após ajustes e correções na topologia do modelo, adição de boca, dentes, língua e olhos, seguiu-se para a etapa de *rigging*.

O processo de *rigging* foi feito em apenas um dos lados do personagem. Os *bones* foram posicionados dentro da malha (Figura 30) e foi implementado o sistema de *IK* nas pernas. Uma vez que o sistema de *IK* estava finalizado e ajustado corretamente em um lado do personagem, a função de simetria foi utilizada para duplicar os mesmos ossos e o sistema de *IK* para o outro lado do personagem.

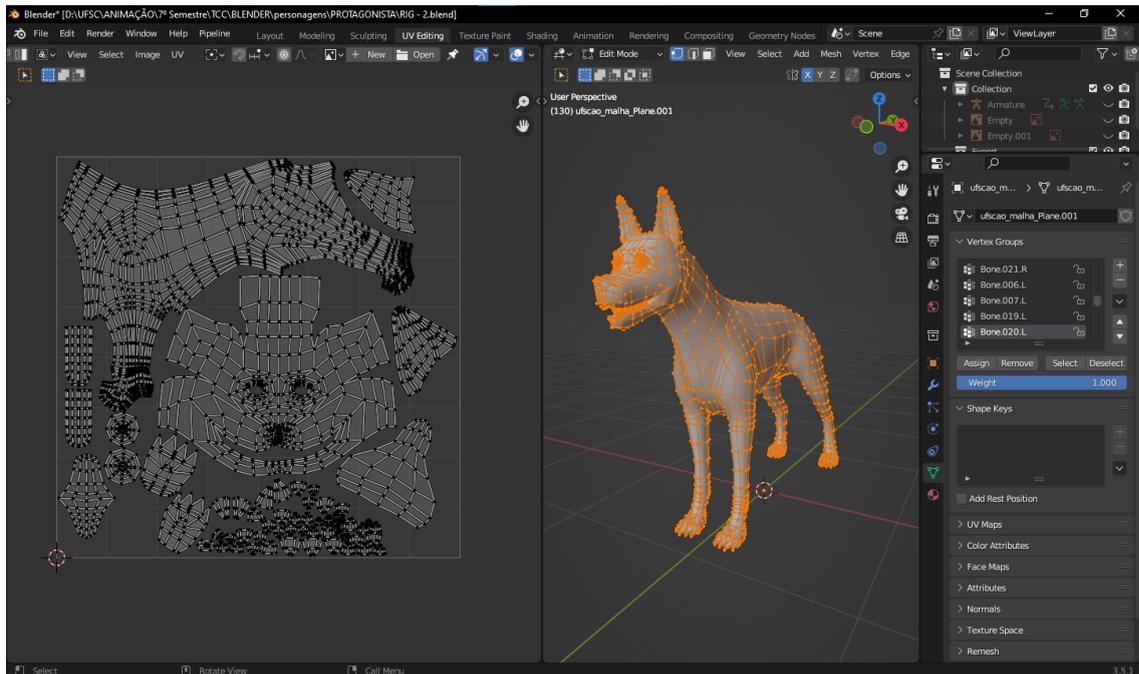
Figura 30 - *Rigging* personagem



Fonte: Autor, 2023.

O processo de *UV mapping* foi realizado manualmente, ou seja, as arestas foram selecionadas e marcadas com *Seam* (costura) para definir quais partes do modelo seriam cortadas para formar o mapa *UV*. O *unwrap* foi feito para criar as ilhas, que são pedaços separados do mapa *UV* representando as superfícies do modelo. As ilhas formadas pelo modelo (Figura 31) foram dimensionadas e escaladas manualmente para garantir que ocupassem o espaço adequado no mapa *UV*.

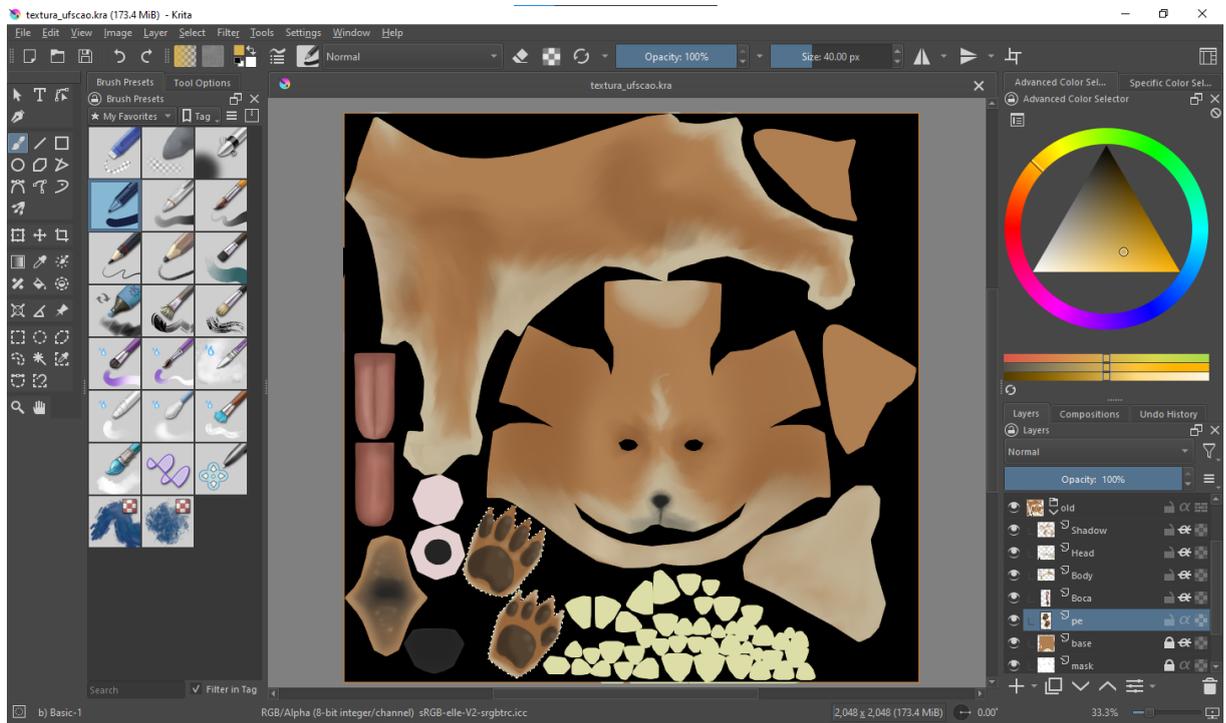
Figura 31 - UV Map personagem



Fonte: Autor, 2023.

Com o *uv map* iniciou-se a texturização do modelo e cada ilha no mapa *UV* foi preenchida com uma cor sólida no *Blender* e teve o mapa exportado como uma imagem para o software de pintura digital *Krita*. Nesse *software*, foi possível utilizar ferramentas de pintura para adicionar elementos visuais e detalhes à textura do modelo, conforme demonstrado na Figura 32.

Figura 32 - Texturização do personagem utilizando o software Krita

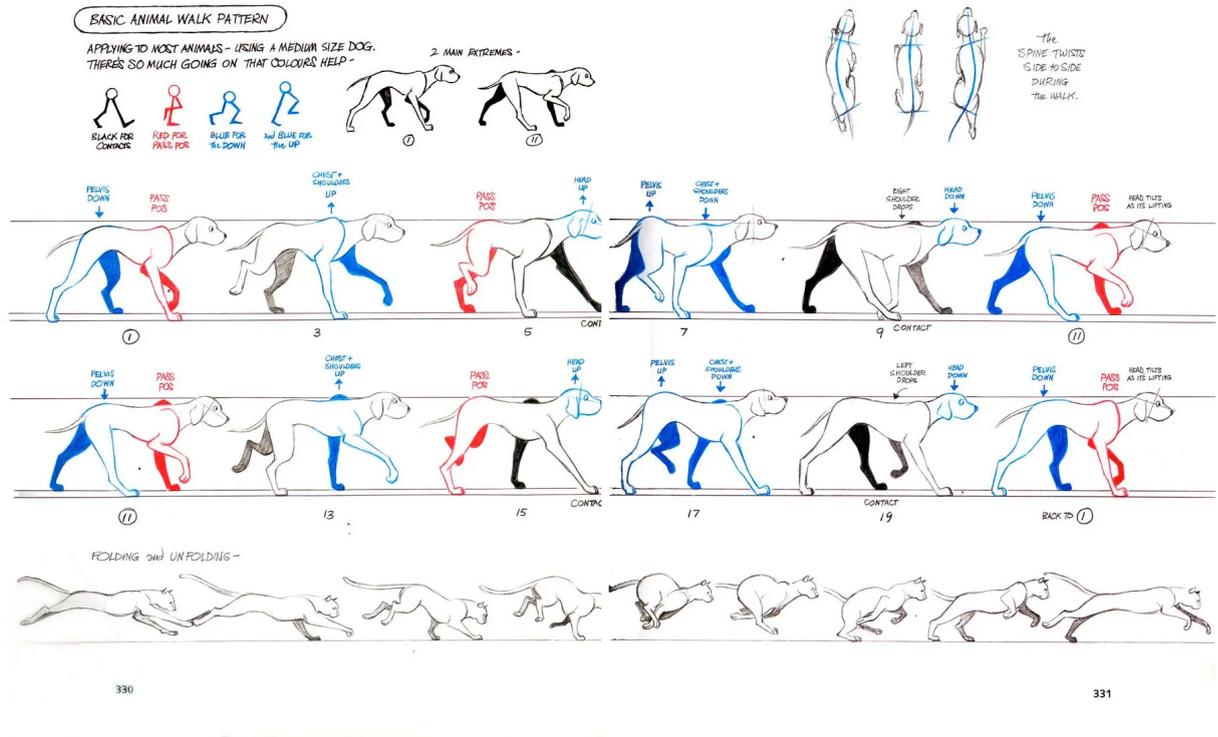


Fonte: Autor, 2023.

Para a animação foram utilizadas duas referências: uma do livro *Animator 's Survival Kit* (WILLIAMS, 2001) que pode ser observada na Figura 33, e outra do vídeo *Animal Gaits for Animators* (STEPHEN CUNNANE, 2017). Dessa forma, foram produzidas 2 animações: *idle* (repouso) e ciclo de caminhada.

As animações foram feitas usando a técnica *pose to pose*, em que foram criadas diversas poses chaves para o personagem que transmitem o movimento desejado, rotacionando e posicionando os *bones* no ambiente 3d. Por causa do escopo de protótipo do projeto, o refinamento da animação não foi executada, mantendo a interpolação linear e sem ajustes. Para exportar para a *engine* foi realizado o *bake*, ou seja, a adição de *keyframes* em toda a timeline, com o intuito de garantir que a mesma animação que estava no *Blender* ia ser reproduzida na *Unreal*, e para remover o *Ik* do personagem.

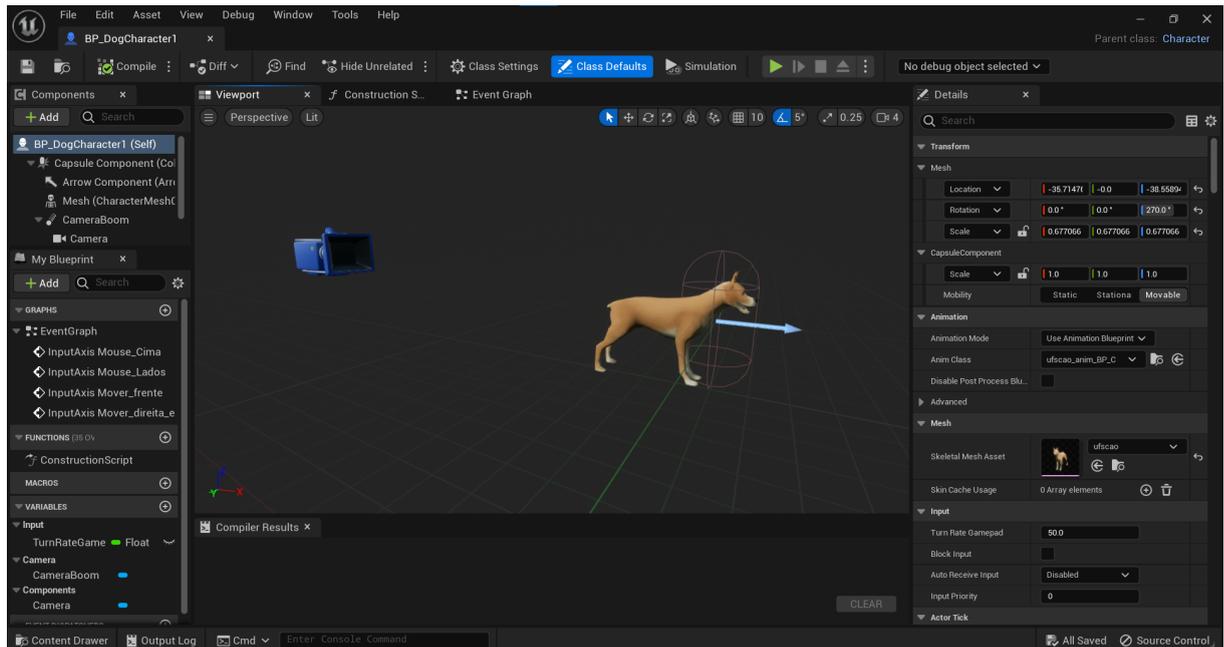
Figura 33 - Referência de ciclo de caminhada para a animação



Fonte: Animator's Survival Kit, Richard Williams, 2016

Para realizar a importação de um arquivo do *Blender* para a *engine*, o arquivo foi salvo no formato *FBX*. Ao enviar para a *engine* foram importadas, a malha, as texturas, o rig, que é conhecido como *skeletal mesh* na *Unreal Engine*, e as animações associadas ao rig. Com todos os elementos importados na *engine*, foi realizado o processo de atualização do modelo na Classe *Blueprint* do personagem principal (Figura 34).

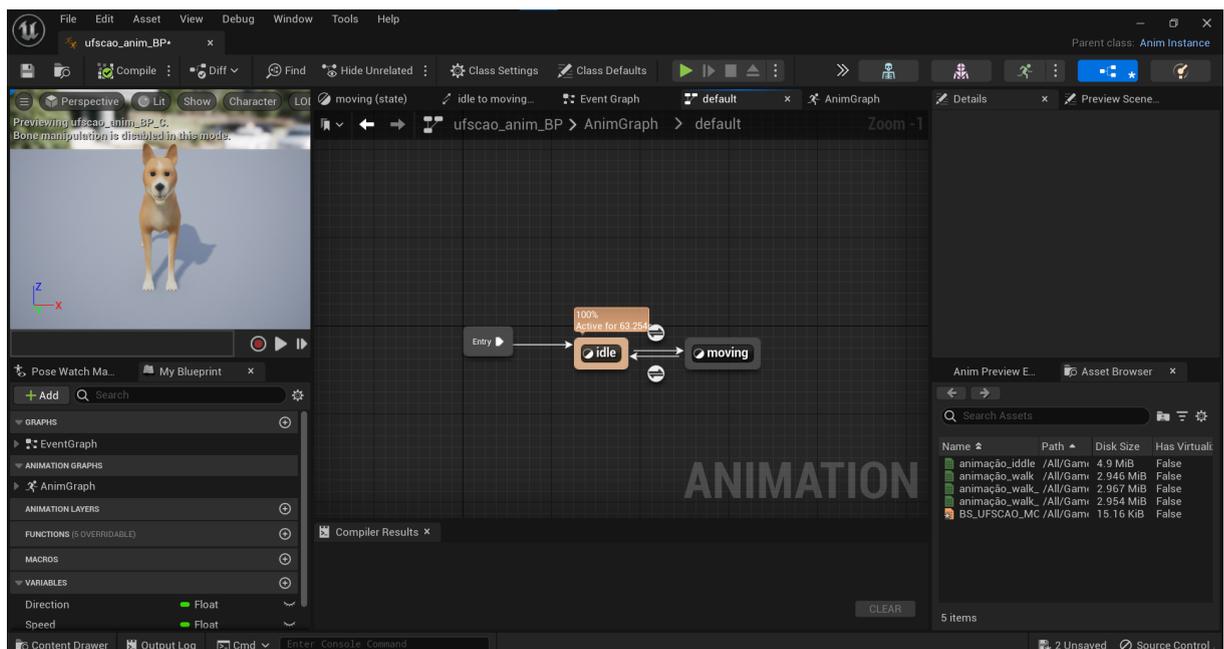
Figura 34 - *Blueprint* do personagem Ufscão na Unreal



Fonte: Autor, 2023.

Para implementar as animações na engine foi necessário criar uma *Animation Blueprint* (Figura 35) e a adição um *Animation State* para gerenciar as transições entre diferentes animações.

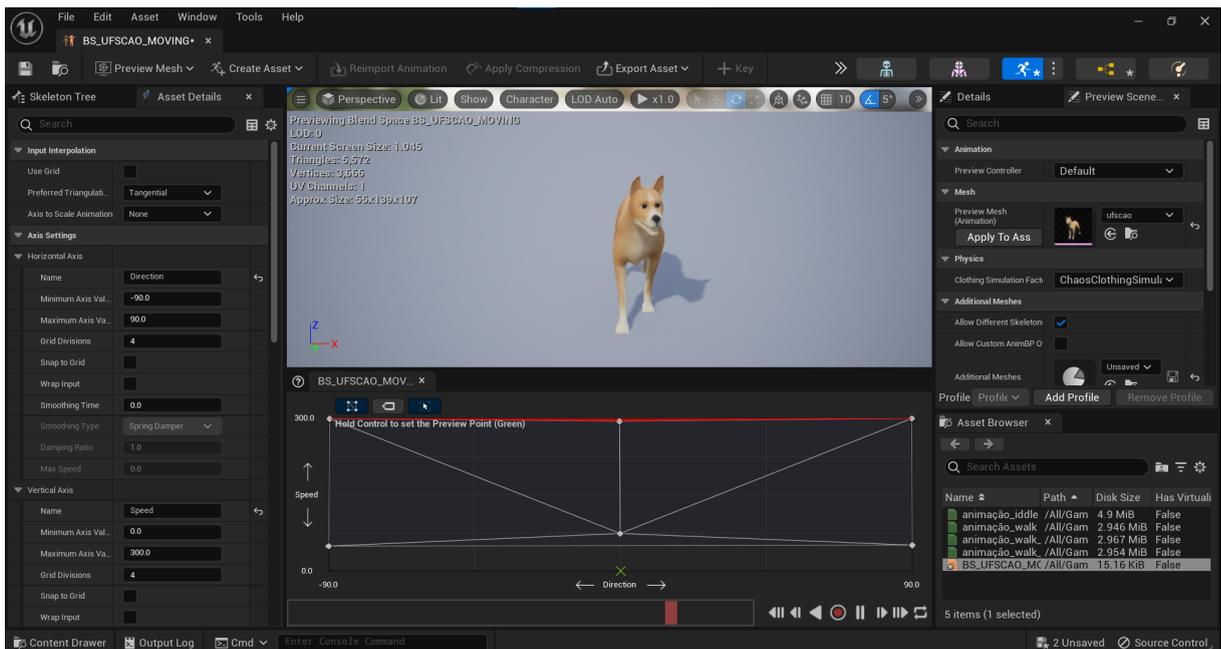
Figura 35 - Tela de *Animation Blueprint*



Fonte: Autor, 2023.

Foram incluídos *nodes* específicos para o movimento em estado de repouso (*idle*) e para a movimentação do personagem. No *node* de movimentação, foi conectado um *Blend Space*, uma função que permite a interpolação entre animações (Figura 36), possibilitando que a animação do personagem também se adaptasse às mudanças de direção.

Figura 36 - Tela de *Blend Space*



Fonte: Autor, 2023.

3.2.3.2 NPC

Os *NPC's (Non-Playable Characters)* são personagens que não podem ser controlados pelos jogadores. No contexto deste protótipo, foram utilizados modelos pré-existentes que já acompanham a engine, os quais possuem animações, rig e texturas próprias. Esses modelos foram empregados com o propósito de preencher o cenário do jogo com mais personagens, adicionando vida e variedade ao ambiente virtual.

3.2.3.3 MASCOTE

O mascote que representa o curso da Animação UFSC é um sagui como mostrado na figura 37, e ele faz parte da identidade visual do curso, sendo produzido pela aluna Luísa Costa em uma competição aberta aos alunos e membros da sociedade em geral.

Figura 37 - Mascote do curso de animação UFSC



Fonte: Luísa Costa

Para a produção do modelo do mascote, foi necessário o ajuste na topologia do modelo, pois em sua primeira versão, a malha apresentava alguns problemas como defeitos na área do nariz e uma quantidade alta de polígonos desnecessários, principalmente no cabelo.

A nova malha foi construída através da extrusão de arestas, em conjunto com a ferramenta de *snap*, que permite posicionar vértices, arestas e planos com maior precisão utilizando as coordenadas de outros objetos como referência. Além disso, o modificador *Shrinkwrap* foi utilizado, permitindo posicionar uma malha em cima da outra como se estivesse sendo embalada a vácuo. O resultado desses ajustes pode ser observado na Figura 38.

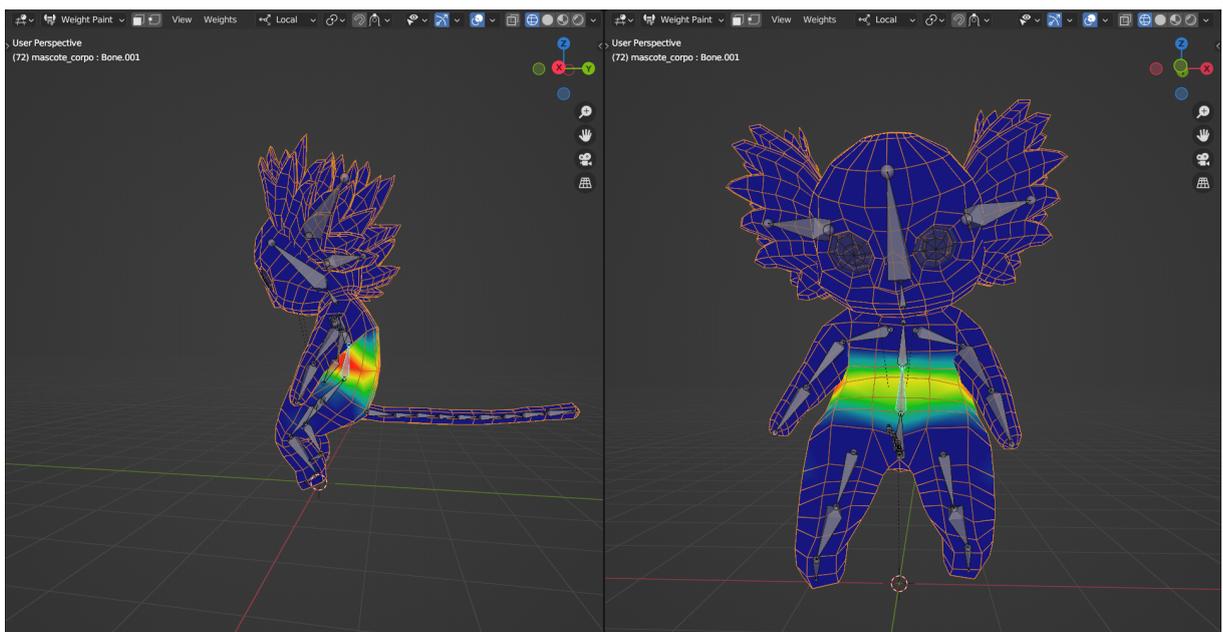
Figura 38 - Processo de retopologia do personagem mascote



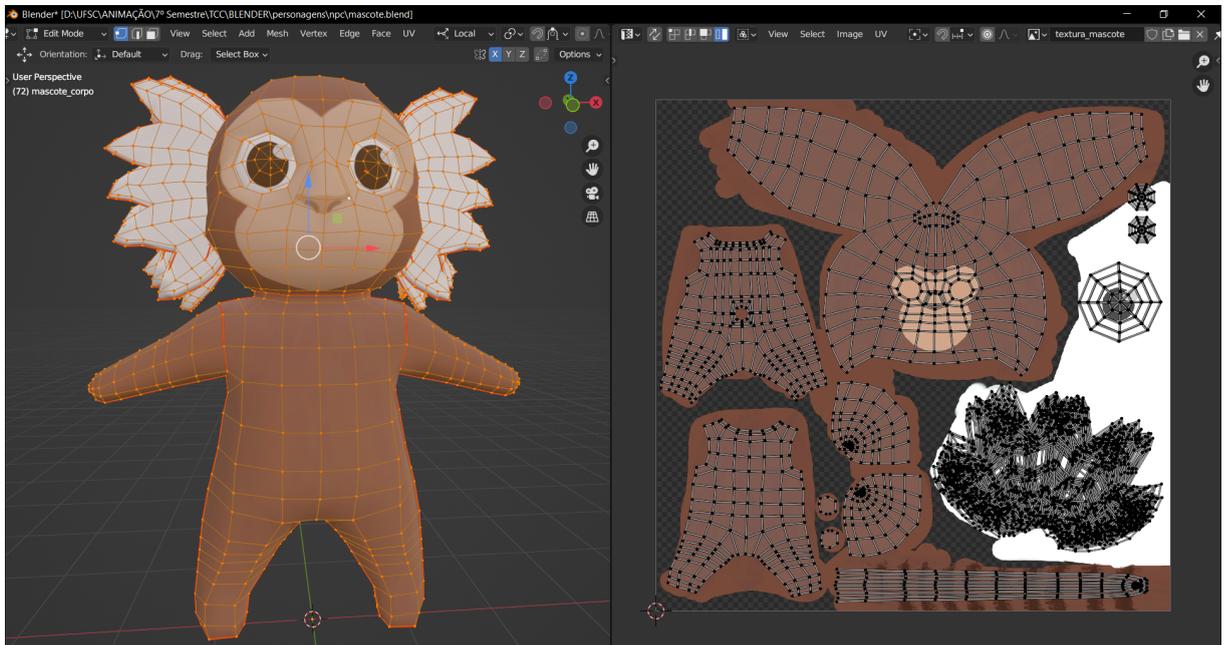
Fonte: Autor, 2023.

As etapas de *unwrap*, textura, *rig* e animação seguiram o mesmo processo do personagem Ufscão, envolvendo ajustes no *weight paint* (Figura 39) e mapeamento da UV por meio de ilhas (Figura 40).

Figura 39 - Processo de ajuste de *weight paint*



Fonte: Autor, 2023.

Figura 40 - *Uv Map* e textura do personagem mascote

Fonte: Autor, 2023.

4 RESULTADOS

A modelagem e textura dos personagens foram consideradas satisfatórias, com destaque para o processo de texturização manual que, apesar de falhas presentes nas áreas em que foi realizado os cortes para o uv map, funcionou bem com o *toon shader*, já que foram apenas adicionada às outlines, conforme demonstrados nas figuras 41 e 42.

Figura 41 - Personagem Ufscão



Fonte: Autor, 2023.

Figura 42 - Personagem mascote



Fonte: Autor, 2023.

Foi utilizado um *asset* disponível gratuitamente no mercado da *epic games*, o *Advanced Cel Shader Lite*, mas houve problemas com o método utilizado para a adição dessas linhas de contorno. Existem locais da malha mais planos que ficam completamente pretos dependendo do ângulo da câmera. Mesmo com esses problemas, o resultado do *outliner* proporciona aos personagens destaque no mundo

do jogo. A implementação das animações para o personagem foi bem-sucedida, com transições suaves entre os estados de repouso e caminhada, mas houve uma falta de refinamento.

O estilo gráfico do cenário não correspondeu às expectativas, eles apresentaram um aspecto mais realista do que o proposto no *Game Design Document*. Seria necessário mais tempo para trabalhar nas texturas e encontrar soluções que permitissem aumentar a estilização desses elementos. Apesar disso, o cenário manteve uma coesão ao adotar um nível de detalhamento e texturas similares entre os objetos.

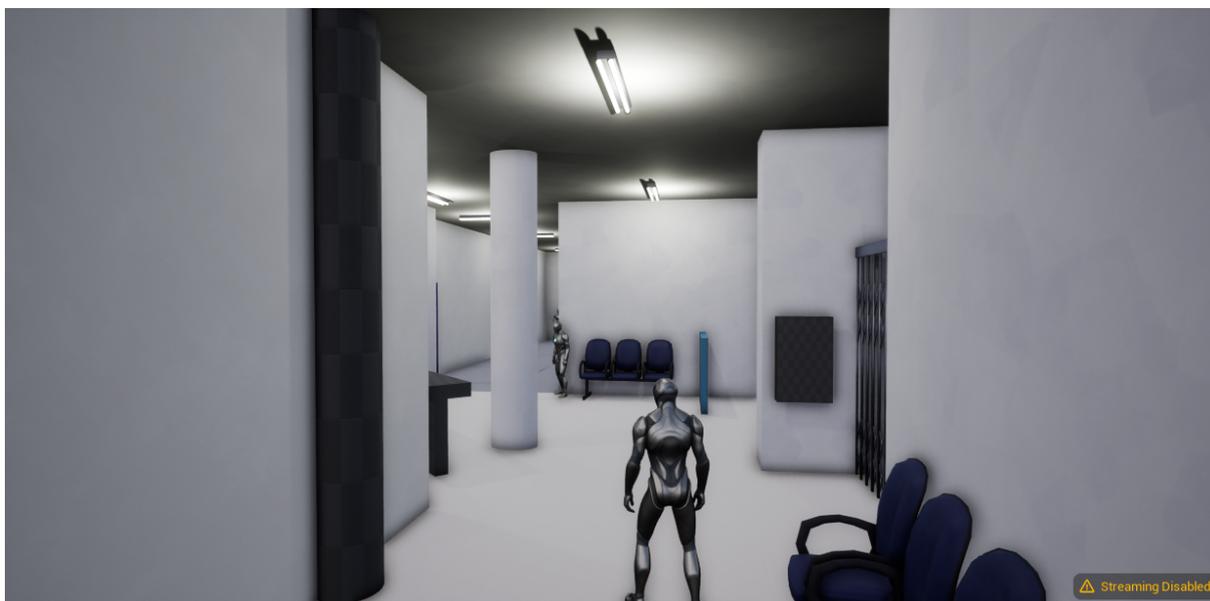
Alguns pontos do mapa foram definidos como de menor importância, estes se encontram em um estado mais inacabado. O cenário com o ponto mais crítico foi o térreo, que no início do projeto seria da região com o maior foco do protótipo e foi a primeira a receber a etapa de bloqueio, mas acabou tendo sua prioridade de finalização abaixada após a definição de como se procederia a apresentação desse protótipo.

Figura 43 - Bloco A, CCE, Térreo



Fonte: Autor, 2023.

Figura 44 - Bloco A, CCE, Térreo em jogo



Fonte: Autor, 2023.

O corredor é a maior área do jogo e, para preencher o local, foram adicionados alguns NPCs pelo cenário.

Figura 45 - Bloco A, CCE, 1º andar.



Fonte: Autor, 2023.

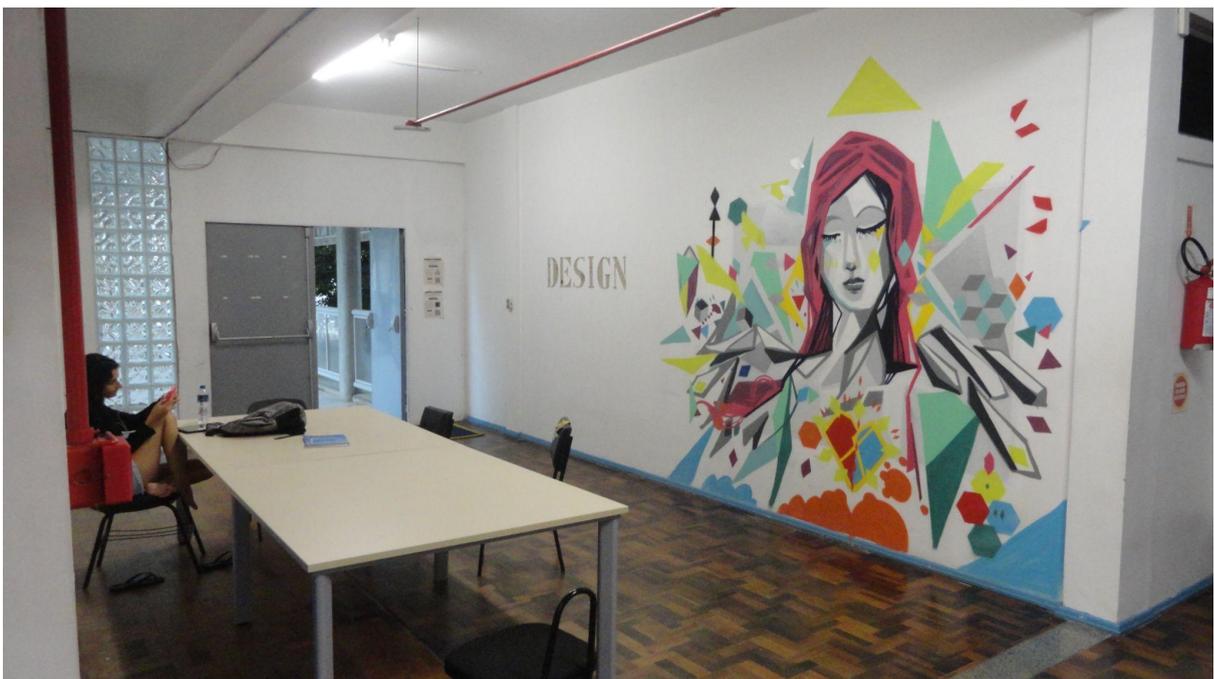
Figura 46 - Bloco A, CCE, 1º andar em jogo.



Fonte: Autor, 2023.

A área existente de ligação entre 2 prédios (figura 47) foi fechada para isolar o jogo em apenas um bloco. Mesmo com medições, algumas áreas ficaram fora de proporção, acarretando em um problema em que não se encontrou uma maneira de alterar a escala das texturas dentro do editor de materiais (Figura 48).

Figura 47 - Bloco A, CCE, 1º andar, parede.



Fonte: Autor, 2023.

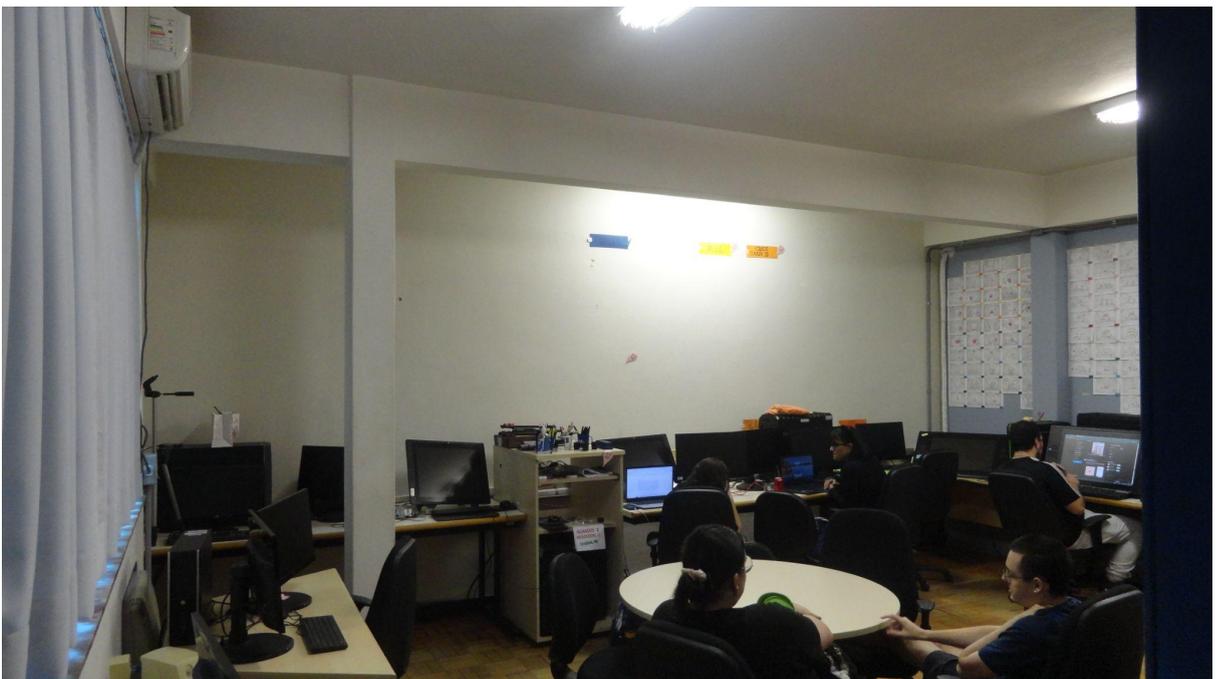
Figura 48 - Bloco A, CCE, 1 andarº, parede em jogo.



Fonte: Autor, 2023.

As salas do *design lab* (figura 50) e sala 130 (figura 51) tiveram o necessário para o reconhecimento delas finalizado.

Figura 49 - Sala Design Lab



Fonte: Autor, 2023.

Figura 50 - Sala Design Lab em jogo



Fonte: Autor, 2023.

As salas fechadas com muitos objetos influenciam negativamente na câmera, que acaba colidindo com eles e causando movimentos muito bruscos na imagem. Pretende-se programar futuramente uma câmera mais suave ou manter ela travada em um ponto específico do mapa ao entrar em salas.

Para este projeto não utilizou-se o sistema de *Motion blur*, uma técnica que borra a câmera ao se movimentar. Ele foi desativado para se ter uma imagem mais limpa e causar menos enjojo em pessoas mais sensíveis ao movimento.

Figura 51 - Sala 130



Fonte: Autor, 2023.

Figura 52 - Sala 130 em jogo



Fonte: Autor, 2023.

Não foram feitas otimizações na unreal e a única técnica para diminuir o número de geometria no cenário foi um sistema de troca de mapa ao entrar nas salas, dessa forma, o computador precisou renderizar apenas uma quantidade de

informações por vez. Em um notebook possuindo uma gtx 1660 TI, um i7 de 9 geração, 16GB de memória RAM, o jogo está rodando em média com uma taxa de 60 *frames* por segundo, mas apresentando gargalos. O protótipo não foi testado em celulares.

5 CONSIDERAÇÕES FINAIS

É evidente a ascendente em que o mercado de *games* se encontra, principalmente a área de jogos *mobile*, que representa cerca da metade do mercado devido à sua acessibilidade proporcionada pelos *smartphones*. Focar nesse nicho em específico é uma estratégia interessante para propiciar uma maior distribuição do jogo em questão.

Os *Serious Games* se destacaram como uma ferramenta promissora no contexto da aprendizagem, principalmente para as gerações mais jovens, que crescem imersas no universo dos jogos eletrônicos e têm maior facilidade em lidar com essa forma de mídia. Acredita-se ser possível criar jogos envolventes que aproveitem essa familiaridade para instruir pessoas em diversos assuntos.

Ao longo do processo de criação do projeto, percebeu-se a amplitude do campo de desenvolvimento de jogos, revelando que, mesmo a construção de um simples protótipo exige conhecimentos de diversas áreas. Trata-se de uma tarefa multidisciplinar que demanda a combinação de habilidades especializadas para alcançar resultados de qualidade. Para a criação dos assets, percebeu-se a necessidade de amplo conhecimento em diferentes ferramentas para a construção de objetos e personagens, abrangendo tarefas como modelagem, mapeamento *UV*, texturização, *rigging* e animação. Além disso, a familiaridade com o sistema de programação baseado em *nodes* dentro da *engine*, revelou-se essencial, tanto para a produção de materiais, quanto para a implementação de diversas funcionalidades no jogo. Algumas áreas da produção como música e sons não foram o foco deste protótipo, outras, como programação e otimização na engine, só foram possíveis graças ao suporte de materiais e tutoriais em plataformas como o *YouTube* e fóruns *online*, que desempenharam um papel indispensável na realização deste projeto, fornecendo esclarecimentos sobre as diversas ferramentas disponíveis nos *softwares* utilizados.

Apesar do escopo inicial do projeto ter considerado as limitações de tempo e falta de conhecimento aprofundado nos *softwares* utilizados, foi necessário abandonar algumas áreas e mecânicas de jogo inicialmente idealizadas devido ao tempo adicional demandado por outras partes essenciais do projeto. As áreas modeladas alcançaram um nível de detalhamento satisfatório em relação ao planejado inicial, capturando as principais características dos locais e apresentando um bom nível de fidelidade em relação aos locais reais.

REFERÊNCIAS

ALVES, F. **Gamification: como criar experiências de aprendizagem engajadoras**. DVS editora, 2015

AMANATIADOU, A.; VAN DE WEERD, I. **Extending the reference method for game production: A situational approach**. 2009 Conference in Games and Virtual Worlds for Serious Applications. IEEE, 2009.

ASLAN, S.; BALCI, O. **GAMED: digital educational game development methodology**. Simulation, v. 91, n. 4, p. 307–319, 2015.

BLENDER FOUNDATION. **Blender 3.4 Manual de referência**. Disponível em: <<https://docs.blender.org/manual/pt/>> Acessado em junho de 2023.

Capture the museum; Edimburgo; Disponível :<<https://www.capturethemuseum.com>> Acessado em junho de 2023.

CHEN, S.; MICHAEL, D. **Serious games: Games that educate, train, and inform**. Florence, AL, USA: Course Technology, 2005.

DROZINA, A ; OREHOVAČKI, T. **Creating a Tabletop Game Prototype in Unreal Engine 4**. Croácia: Juraj Dobrila University of Pula, 2018

EPIC GAMES. **Perguntas frequentes (FAQs)** Disponível :<<https://www.unrealengine.com/pt-BR/faq>> Acessado em junho de 2023.

EPIC GAMES. **UNREAL ENGINE 5 DOCUMENTATION** Disponível :<<https://docs.unrealengine.com>> Acessado em junho de 2023.

Family Quest; Amsterdam; Disponível :<<https://www.rijksmuseum.nl/en/whats-on/families-children/family-game>> Acessado em junho de 2023.

FORTIM, Ivelise (Org). **Pesquisa da indústria brasileira de games 2022**. ABRAGAMES: São Paulo, 2022. Disponível em: <<https://www.abragames.org/uploads/5/6/8/0/56805537/abragames-pt.pdf>> Acessado em junho de 2023.

INÁCIO, L.R. **Desenvolvimento de um Plug-In de Espelhamento de Animações Tridimensionais para o Motor Unreal Engine 4**

. Araranguá: Universidade Federal de Santa Catarina, 2022.

JAMES, B. UV Mapping Explained. 2012. Disponível em: <<https://worldorigin.wordpress.com/2012/05/30/uv-mapping-explained-36/>> Último acesso em: junho de 2023.

LUQUE, R. R. **The Cel Shading Technique**. , 2012. Disponível em: <<https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=4134d230132b3639aef4c8e05557beea8c94ae81> > Acessado em junho de 2023.

MATTAR, J. **Games em Educação: como os nativos digitais aprendem**. São Paulo: Pearson Prentice Hall, 2010.

MOTTA, R. L.; JUNIOR, J. T. **Short game design document (SGDD)**. , 2013. Disponível em: <https://www.sbgames.org/sbgames2013/proceedings/artedesign/15-dt-paper_SGD_D.pdf > Acessado em junho de 2023.

NETO, A. F. **DESENVOLVIMENTO DE PERSONAGEM PARA JOGOS ELETRÔNICOS 3D : CRIAÇÃO E CONSTRUÇÃO ADAPTADAS AO CONTEXTO NACIONAL DE PRODUÇÃO**. Florianópolis: Universidade Federal de Santa Catarina, 2017.

NEWZOO. **Key Insights into Brazilian Gamers | report**. Newzoo, San Francisco, 22 Set. 2022. Disponível em: <<https://newzoo.com/resources/trend-reports/key-insights-into-brazilian-gamers-newzoo-gamer-insights-report>> Acessado em junho de 2023.

NEWZOO. **Global Games Market Report 2022 | Free Version**. Newzoo, San Francisco, 26 Jul. 2022. Disponível em: <<https://newzoo.com/resources/trend-reports/newzoo-global-games-market-report-2022-free-version>> Acessado em junho de 2023.

NEWZOO. **Global Games Market Report 2022 | Free Version**. Newzoo, San Francisco, 16 Mar. 2023. Disponível em: <<https://newzoo.com/resources/trend-reports/pc-console-gaming-report-2023>> Acessado em junho de 2023.

PESQUISA GAME BRASIL 2023, Disponível :<www.pesquisagamebrasil.com.br> Acessado em junho de 2023.

PIVEC, M.; KRONBERGER, A. **Virtual museum: Playful visitor experience in the real and virtual world**. 2016 8th International Conference on Games and Virtual Worlds for Serious Applications (VS-GAMES). IEEE, 2016.

RAMADAN, R.; WIDYANI, Y. **Game development life cycle guidelines**. 2013 International Conference on Advanced Computer Science and Information Systems (ICACSIS) .IEEE, 2013.

ROCHA, L. L. R. **"A indústria criativa de games no Brasil"** . 2015.

SALUTES, B. **O que é motor gráfico?**. 2022. Disponível em: <<https://canaltech.com.br/software/o-que-e-motor-grafico/>> Último acesso em: junho de 2023.

TORRES-FERREYROS, C. M.; FESTINI-WENDORFF, M. A.; SHIGUIHARA-JUAREZ, P. N. **Developing a videogame using unreal engine based on a four stages methodology**. 2016 IEEE ANDESCON. IEEE, 2016.

VENUTA, A. **Implementing Skeletal Animation**. 2014. Disponível em: <<https://veenu.github.io/blog/implementing-skeletal-animation/>> Último acesso em: junho de 2023.

WARDRIP-FRUIIN, N.; MONTFORT, N. **The new media reader**. MIT press, 2003.

WILLIAMS, R. **Manual de animação: Manual de métodos, princípios e fórmulas para animadores clássicos**, de computador, de jogos de stop motion e de internet. São Paulo: Editora Senac São Paulo, 2016

XIMENES, L. **O que é motor gráfico?**. 2022. Disponível em: <<https://www.hardware.com.br/artigos/o-que-e-motor-grafico/>> Último acesso em: junho de 2023.