



Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"

RAFAEL FIGUEIREDO ZANETTI

NOVA VENTIS ADVENTURERS: A aplicação da Realidade Aumentada e
Gamificação em um Jogo.

Assis/SP

2018



Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"

RAFAEL FIGUEIREDO ZANETTI

**NOVA VENTIS ADVENTURERS: A aplicação da Realidade Aumentada e
Gameficação em um Jogo.**

Trabalho de conclusão de curso apresentado ao curso de Bacharelado em Ciência da Computação do Instituto Municipal de Ensino Superior de Assis – IMESA e a Fundação Educacional do Município de Assis – FEMA, como requisito parcial à obtenção do Certificado de Conclusão.

Orientando: Rafael Figueiredo Zanetti
Orientador: Prof. Esp. Célio Desiró

Assis/SP

2018

FICHA CATALOGRÁFICA

ZANETTI, Rafael.

Nova Ventis Adventurers: A aplicação da Realidade Aumentada e Gamificação em um Jogo / Rafael Figueiredo Zanetti. Fundação Educacional do Município de Assis – FEMA – Assis, 2018.

78p.

1. Realidade Aumentada (RA). 2. Gamificação. 3. Unity. 4. Vuforia

CDD: 001.42
Biblioteca da FEMA

NOVA VENTIS ADVENTURERS: A aplicação da Realidade Aumentada e
Gameificação em um Jogo.

RAFAEL FIGUEIREDO ZANETTI

Trabalho de Conclusão de Curso
apresentado ao Instituto Municipal de
Ensino Superior de Assis, como requisito do
Curso de Graduação, avaliado pela seguinte
comissão examinadora:

Orientador: Prof. Esp. Célio Desiró

Examinador: Prof. Dr. Osmar Aparecido Machado

RESUMO

Este projeto procura, de forma funcional, unir os conceitos de gamificação às tecnologias de criação de jogos da ferramenta Unity, aliadas ao Vuforia, uma SDK, kit de desenvolvimento de software, para o estudo e desenvolvimento do conceito de um jogo mobile com realidade aumentada que, através de suas mecânicas e características, traga um maior engajamento e recompensa ao jogador em seu autodesenvolvimento e interação social. Apresentando as aplicações já vistas sobre os temas aqui abordados e, assim, criando um estudo e uma nova perspectiva sobre o desenvolvimento e utilização deste tipo de aplicação, técnicas, conceitos e mecânicas envolvidas.

Palavras-chave: Realidade Aumentada (RA), Gamificação, Unity, Vuforia.

ABSTRACT

This project seeks, in a functional way, to link game concepts to the game creation technologies of the Unity tool, allied to Vuforia, an SDK, a software development kit, for the study and development of the concept of a mobile game with augmented reality that, through its mechanics and characteristics, brings a greater commitment and reward to the player in its self-development and social interaction. Introducing the applications already seen on the themes addressed here, and thus creating a study and a new perspective on the development and use of this type of application, techniques, concepts and mechanics involved.

Keywords: Augmented Reality (AR), Gamification, Unity, Vuforia.

LISTA DE ILUSTRAÇÕES

Figura 1: Interface do editor de cenas da Unity3D.	21
Figura 2: Interface da janela Project.....	22
Figura 3: Interface da janela Hierarchy.	22
Figura 4: Interface da janela Scene.....	23
Figura 5: Interface da janela Inspector.	24
Figura 6: Interação de um Modelo Target e seu objeto real.....	25
Figura 7: Objeto projetado sobre um Image Target.....	26
Figura 8: Objeto Multi-Target e suas dimensões.....	26
Figura 9: Projeção de um Cylinder Target em um objeto real.	27
Figura 10: Teste de um Object Target em um celular.	28
Figura 11: Um VuMark sendo escaneado e mostrando suas informações.	28
Figura 12: Demonstração da projeção de um objeto em Ground Plane.....	29
Figura 13: Tela de adição de chave de licença do Vuforia.	30
Figura 14: Tela de visualização da chave de licença do Vuforia.	30
Figura 15: Configuração da instancia do Vuforia no projeto Unity.	31
Figura 16: Criação de um pote a partir da deformação de um objeto.	33
Figura 17: Diagrama de representação do funcionamento básico.	36
Figura 18: Demonstração da aplicação exemplo.	37
Figura 19: Configuração do DroneTarget.	38
Figura 20: Configuração do GoalTarget.	38
Figura 21: Configuração do objeto Drone.....	39
Figura 22: Visualização do NavMesh.	40
Figura 23: Criação do objeto Ring.....	40
Figura 24: Declaração das variáveis de referencia dos objetos.	41
Figura 25: Inicialização da variável agent.	41
Figura 26: Função de movimentação do Drone.	42
Figura 27: Configuração do scrpt de movimentação.	42
Figura 28: Código da função MapRange.....	43
Figura 29: Função de controle da inclinação do Drone.	43
Figura 30: Configuração da sombra do Drone.	44
Figura 31: Configurando a linha.	44
Figura 32: Configuração do Line Render dentro do objeto Ring.	45
Figura 33: Código da função Goal.....	45
Figura 34: Criação e visualização do objeto Destination.....	46
Figura 35: Modificação na função de movimentação.	46

SUMÁRIO

1	INTRODUÇÃO	11
1.1	OBJETIVO	13
1.2	PÚBLICO ALVO	13
1.3	JUSTIFICATIVA	13
1.4	MOTIVAÇÃO.....	14
1.5	PERSPECTIVAS DE CONTRIBUIÇÃO	15
1.6	MÉTODO DE DESENVOLVIMENTO.....	15
1.7	RECURSOS.....	16
2	REVISÃO BIBLIOGRÁFICA.....	17
2.1	ANDROID.....	17
2.2	SQLITE	18
2.3	UNITY	19
2.3.1	Interface e utilização.....	21
2.4	VUFORIA	24
2.4.1	Vuforia e a Unity	29
2.5	GEOLOCALIZAÇÃO	32
2.6	BLENDER	33
2.7	GAMIFICAÇÃO.....	34
3	PROPOSTA DE TRABALHO	36
4	APLICANDO A RA	37
5	GAMIFICANDO.....	47
5.1	MECÂNICAS.....	47
5.2	ATRIBUTOS IN-LIFE	48
5.2.1	Força	48
5.2.2	Destreza	48
5.2.3	Agilidade.....	49
5.2.4	Constituição.....	49

5.2.5	Inteligência	49
5.2.6	Sabedoria	49
5.2.7	Carisma	50
5.2.8	Criatividade.....	50
5.2.9	Relacionamento entre os atributos	50
5.3	ATIVIDADES.....	50
5.4	GRUPOS E A COMUNIDADE	51
5.4.1	GRUPOS	51
5.4.2	COMUNIDADE	52
5.4.2.1	Plaza	52
5.4.2.2	Fórum.....	53
6	O JOGO	54
6.1	HISTÓRIA	54
6.2	COMEÇANDO A JOGAR.....	54
6.3	OS PERSONAGENS	55
6.3.1	Atributos	56
6.3.1.1	Força.....	56
6.3.1.2	Destreza.....	56
6.3.1.3	Agilidade	56
6.3.1.4	Constituição	57
6.3.1.5	Inteligência	57
6.3.1.6	Sabedoria.....	57
6.3.1.7	Carisma.....	57
6.3.1.8	Criatividade	57
6.3.2	Raça	58
6.3.2.1	Humanos.....	58
6.3.2.2	Elfos	58
6.3.2.3	Anões.....	59
6.3.3	Árvores de habilidades	60
6.3.4	Perícias.....	61
6.3.4.1	Valores de perícia	63

6.3.4.2	Perícias em prática	63
6.3.5	Talentos.....	64
6.4	MECÂNICAS IN-GAME.....	65
6.4.1	Mundos e exploração	65
6.4.2	Estatísticas de personagem	67
6.4.2.1	Evolução do personagem.....	69
6.4.3	Combate.....	70
6.4.3.1	Ataques e dano	71
6.4.3.2	Magias.....	71
6.4.4	Monstros.....	72
6.4.5	Non Player Characters	72
6.4.6	Aventurando-se	73
6.4.7	Itens.....	73
6.4.8	Regras.....	73
7	CONSIDERAÇÕES FINAIS	75
8	REFERÊNCIAS.....	76

1 INTRODUÇÃO

Com a evolução das tecnologias de hardware, software e comunicação digital, temos disponíveis mais formas de interação com as informações e dados virtuais existentes em nossa realidade. E é nisso que se baseiam as tecnologias de Realidade Aumentada (RA) e Realidade Virtual (RV), as interações de um ambiente virtual em nosso ambiente real, no caso da RA, e o inverso, no caso da RV.

Atualmente, a RA, junto da RV, vêm sendo aplicadas nas mais diversas áreas de conhecimento e do cotidiano, como na medicina, entretenimento, marketing e aprendizado. Isso se dá através de interações únicas com objetos virtuais e/ou informações, inseridas em nossa realidade, através das tecnologias de RA. Por exemplo, na área de medicina, tecnologias de RA vêm sendo usadas para visualização dos vasos sanguíneos, em tempo real, de um paciente. O artigo “Estado da Arte Brasileira sobre a Realidade Aumentada” aborda um exemplo de aplicação de RA à realidade médica: “com auxílio de uma câmera especial que capta o reflexo e envia-o a um computador, que processa as informações e gera uma imagem virtual das veias sobre a pele, a RA proporciona aos médicos verem as veias problemáticas e basta um disparo de laser na região para curar a veia, eliminando assim a necessidade de cirurgias, que sempre acabam por limitar temporariamente certas atividades dos operados”.

Na área de entretenimento, os maiores exemplos são os jogos, que levam a interação do jogador a outro nível com a aplicação das tecnologias de RA. Recentemente, o jogo que mais esteve em evidência foi o “Pokémon GO”, no qual o jogador interage, por meio de seu celular, com objetos virtuais, posicionados em pontos geográficos, através do georreferenciamento, em nossa realidade.

Ainda, decorrente da área dos jogos, vemos o crescimento da gamificação da vida, na qual se aplicam conceitos de jogos à vida real, buscando trazer melhores resultados, maior engajamento e interesse do usuário sobre a atividade “gamificada”. Isso se dá através da implementação de estruturas de desafios e recompensas, utilizando-se ou não de meios virtuais para o mesmo. Como apontado no artigo “Gamificação: a transformação do conceito do termo jogo no contexto da pós-modernidade” (Gabrielle Navarro, 2011), uma escola de Nova York tem aplicado conceitos de gamificação em suas aulas presenciais, para “estimular a necessidade de aprender, motivando e criando no estudante o interesse em adquirir conhecimento”. A gamificação vem sendo aplicada aos mais diversos âmbitos sociais, acadêmicos e profissionais. Como também citado no mesmo artigo de Gabrielle Navarro, a Petrobras tem usado um simulador, ou seja, um jogo, para melhorar a instrução de seus funcionários na manipulação correta de seus equipamentos e evitar acidentes decorrentes da má manipulação.

Com a aplicação dessas tecnologias e conceitos, aliados ao desenvolvimento mobile, este projeto busca o desenvolvimento de um jogo multiplataforma com RA, explorando os conceitos da gamificação a suas interações com os jogadores e proporcionando uma experiência nova e diferente das propostas atuais presentes no mercado, unindo a experiência real com a virtual, para que, por meio dos conceitos da gamificação, possa recompensar o jogador pelas atividades, tanto virtuais, quanto reais, de forma a despertar maior interesse, não apenas no seu desenvolvimento no jogo, mas também com seu desenvolvimento real.

Para isso, será utilizado o Vuforia, um Kit de desenvolvimento de software (SDK), que utiliza de rastreamento e reconhecimento de superfícies, marcadores e objetos 3D, para localizar e reproduzir os objetos virtuais naquela posição real.

1.1 OBJETIVO

Este projeto tem como objetivo desenvolver uma pesquisa para aplicação da gamificação de novas formas em um jogo baseado no gênero de RPG, com as seguintes características: conexão online e suporte para múltiplos jogadores em rede, interações únicas utilizando-se da RA e com a aplicação de conceitos da gamificação, possa recompensar o jogador por suas atividades, tanto virtuais, quanto reais, de forma a despertar maior interesse, não apenas no seu desenvolvimento no jogo, mas também com seu desenvolvimento pessoal, incluindo o âmbito social. Para isso, serão utilizadas a Unity, com aplicações de RA por meio do Vuforia.

1.2 PÚBLICO ALVO

O produto do desenvolvimento deste projeto tem como público alvo jovens, e até mesmo adultos que tenham interesse por este tipo de produto. Tendo enfoque a usuários que estão à procura de uma nova abordagem ao mundo dos jogos e também algo que os incentive a realizar determinadas tarefas na realidade.

Além da comunidade e desenvolvedores que possam necessitar de base e novas perspectivas para o desenvolvimento de mecânicas de gamificação e como aplica-las a um jogo.

1.3 JUSTIFICATIVA

Este projeto possibilitará maiores estudos e perspectivas de aplicações da RA e da gamificação, tanto na vida cotidiana dos usuários, quanto no próprio jogo, além de proporcionar uma nova experiência e apresentar novos padrões de

desenvolvimento e aplicação de mecânicas nos jogos de RPG, resgatando antigos conceitos e os renovando para tal objetivo.

Buscando, também, suprir a demanda crescente de aplicações desenvolvidas em ambientes mais interativos, seja virtual, seja real, que possam trazer novas interações aos usuários, despertando maior engajamento em suas atividades.

1.4 MOTIVAÇÃO

A proposta deste desenvolvimento se deu a partir da necessidade do estudo de novas aplicações e perspectivas do uso da RA, do desenvolvimento e aplicações das ferramentas Vuforia e Unity, juntamente a um estudo sobre as aplicações da gamificação da vida em um jogo que utiliza RA.

Assim, este projeto busca o desenvolvimento de um jogo multiplataforma com RA, que possa aplicar os conceitos da gamificação a suas interações com os jogadores e proporcionar uma experiência nova e diferente das propostas presentes no mercado. Para isso, utilizando-se o Vuforia, um Kit de desenvolvimento de software, junto da Unity, um ambiente de desenvolvimento de jogos multiplataforma compatível com o Vuforia, que possibilita o desenvolvimento de jogos *mobile* (para celulares e *tablets*) com aplicação de RA.

Unindo assim, a experiência real com a virtual, para que, através dos conceitos da gamificação, o jogador possa ser recompensado por suas atividades, tanto virtuais, quanto reais, de forma a despertar maior interesse, não apenas no seu desenvolvimento no jogo, mas também em seu desenvolvimento real.

1.5 PERSPECTIVAS DE CONTRIBUIÇÃO

Com o desenvolvimento do “Nova Ventis Adventurers”, a área de jogos e desenvolvimento em RA poderá ter novos padrões de desenvolvimento nos quais se baseiam novas visões sobre a utilização das tecnologias disponíveis, uma aproximação da união de gamificação da vida e os jogos virtuais, e um estudo dessa união e suas aplicações usadas neste projeto. Também buscando suprir um nicho ainda vago presente nesta plataforma.

1.6 MÉTODO DE DESENVOLVIMENTO

Para a elaboração desse projeto de conclusão de curso serão utilizados artigos, fóruns, sites, livros e manuais de utilização disponibilizados pelas equipes de desenvolvimento da Unity e do Vuforia, que forneceram informações e métodos para o desenvolvimento da aplicação mobile proposta.

O desenvolvimento do projeto se dará através da utilização da IDE Unity ao desenvolvimento de jogos, aliado ao Vuforia, para a criação e aplicação das funções de RA. Utilizando, também, ferramentas de design de modelos 3D e 2D para as diversas funções da aplicação.

Na primeira etapa, serão pesquisadas maiores informações sobre as ferramentas e conceitos a serem utilizados, tanto quanto a aprendizagem e aprimoramento do uso das mesmas.

Na segunda etapa, será desenvolvida uma aplicação com as informações levantadas e habilidades desenvolvidas na etapa anterior.

Na terceira etapa, o projeto será apresentado a usuários de teste, que opinarão e apresentarão pontos positivos e negativos da aplicação, os quais serão analisados e levarão a modificações no projeto.

1.7 RECURSOS

Para o desenvolvimento do projeto, utilizando dos conceitos apresentados nesta pesquisa, fazem-se necessários os seguintes recursos:

- Um computador que suporte a Unity, com recursos gráficos necessários para a modelagem 3D e poder de processamento o suficiente para a compilação e testes dos códigos na Unity;
- Um dispositivo mobile que tenha suporte aos recursos necessário para a utilização do Vuforia;
- Última versão disponível do software Unity;
- Última versão disponível do SDK Vuforia;
- Banco de Dados SQLite;
- Software de design de modelos 3D e 2D;
- Acesso a livros, sites, fóruns e a documentação disponibilizada sobre as ferramentas utilizadas;
- Acesso a sites, fóruns e artigos necessários para o desenvolvimento do projeto.

2 REVISÃO BIBLIOGRÁFICA

Neste capítulo serão introduzidas as tecnologias e conceitos que serão utilizadas no projeto, abordando suas características e seus conceitos gerais de funcionamento.

O Android será a plataforma para a execução da aplicação a ser desenvolvida nesta proposta. O SQLite é a tecnologia de banco de dados que será utilizada para armazenar e acessar os dados do aplicativo.

O Unity e Vuforia serão responsáveis pelo desenvolvimento do jogo, suas funcionalidades, mecânicas e design.

Os conceitos de gamificação serão introduzidos às interações virtuais e reais, e à lógica de jogabilidade do jogo.

2.1 ANDROID

O Android é um *stack* software, um composto de diversos *softwares* que trabalham juntos para realizar um objetivo, de código aberto para uma ampla gama de dispositivos móveis, sendo um projeto de código aberto liderado pelo Google (Android Source, 2018) que, como destacado por Lars Vogel, no artigo "*Android development tutorial*", o Android possui um sistema operacional baseado em Linux com uma interface programada em Java, sendo que estas características são responsáveis por sua versatilidade e adaptabilidade, podendo até mesmo importar bibliotecas externas referenciadas em seu código.

Outras características citadas por Vogel em seu artigo são:

"[...] permite o processamento em segundo plano, fornece uma biblioteca de interface de usuário rica, suporta gráficos 2-D e 3-D usando as

bibliotecas OpenGL, acesso ao sistema de arquivos e fornece um SQLite incorporado à base de dados. Os aplicativos Android consistem em diferentes componentes e podem reutilizar componentes de outras aplicações. Isso leva ao conceito de uma tarefa no Android; um aplicativo pode reutilizar outros componentes Android para arquivar uma tarefa”.

2.2 SQLITE

“O SQLite é um banco de dados relacional embutido de código aberto. Originalmente lançado em 2000, foi projetado para fornecer uma maneira conveniente para aplicativos gerenciarem dados sem a sobrecarga que muitas vezes vem com sistemas dedicados de gerenciamento de banco de dados relacional. SQLite tem uma reputação por ser altamente portátil, fácil de usar, compacto, eficiente e confiável”(OWENS; ALLEN, 2010).

Assim, o banco SQLite não tem um instância única, ele é embutido dentro do aplicativo ao qual serve, criando uma base de dados para este software. Eliminando a necessidade de conexão ou configuração de um banco de dados externo.

O SQLite também possui diversas bibliotecas ou APIs para o Android, a fim de melhorar seu processamento, entendimento e/ou uso, como no caso da biblioteca *Room Persistence*, desenvolvida pela Google. Conforme apontado pela documentação do Android, na seção *“Room Persistence Library”*: “[...] essa API fornece uma camada de abstração sobre SQLite para permitir o acesso fluente de banco de dados enquanto aproveita todo o poder do SQLite. A biblioteca ajuda a criar um *cache* dos dados do aplicativo em um dispositivo que está sendo executado. Este *cache*, que serve como fonte única do software, permite que os usuários vejam uma cópia consistente das informações-chave no seu aplicativo, independentemente se os usuários possuírem uma conexão com a Internet” (Android Developers, 2018).

2.3 UNITY

Unity, também conhecido como Unity 3D, é um motor de jogo 3D proprietário e uma IDE criada pela Unity Technologies. Conforme apontado pelo estudo “Tutorial: Desenvolvimento de Jogos com Unity 3D” (PASSOS et al , 2007), publicado no “VIII Simpósio Brasileiro de Jogos e Entretenimento Digital”: “A Unity3D abstrai do desenvolvedor de jogos a necessidade de utilizar diretamente DirectX ou OpenGL (apesar de ainda ser possível, caso necessário), suportando a criação de *Shaders* complexos com a linguagem Cg da NVidia. Internamente, o subsistema de simulação física é o popular PhysX, também da NVidia. Para a execução de *scripts*, a Unity usa uma versão de alto desempenho da biblioteca Mono, uma implementação de código aberto do *framework* .Net da Microsoft”.

Outra característica da Unity é que ela possui suporte a plug-ins externos, estes sendo divididos em duas categorias “*Native Plugins*” e “*Managed Plugin*”, segundo o manual da Unity, as diferenças entre esses plug-ins são:

- *Managed Plugins*: “Os *Managed Plugins* (plug-ins gerenciados) são *managed .NET assemblies* criados com ferramentas como Visual Studio ou MonoDevelop. Eles contêm somente o código .NET, o que significa que eles não podem acessar recursos que não sejam suportados pelas bibliotecas .NET. No entanto, o código gerenciado é acessível às ferramentas padrão .NET que a Unity usa para compilar *scripts*. Há, portanto, pouca diferença no uso entre o código de plugin gerenciado e o código de *script* Unity, exceto pelo fato dos plugins serem compilados fora de Unity e, portanto, a fonte pode não estar disponível” (Manual Unity, 2018).
- *Native Plugins*: “Os *Native Plugins* (plug-ins nativos) são bibliotecas de código nativo específicas da plataforma. Eles podem acessar recursos

como chamadas de SO (sistema operacional) e bibliotecas de códigos de terceiros que, de outra forma, não estarão disponíveis para Unity, ao contrário das bibliotecas gerenciadas. Por exemplo, se esquecer-se de adicionar um arquivo de plug-in gerenciado ao projeto, receberá mensagens de erro padrão do compilador. Se fizer o mesmo com um plug-in nativo, só verá um relatório de erro quando tentar executar o projeto” (Manual Unity, 2018).

Desta forma se dão suas ligações e comunicações com o SDK Vuforia, e muitos outros. Além disso, a Unity possui suporte para múltiplas plataformas, atualmente contando com aproximadamente 25 plataformas disponíveis para desenvolvimento, entre elas o Android e o iOS. Sendo assim, um mesmo projeto criado em Unity pode ser traduzido em diversas plataformas que ele suporte e tenham os recursos necessários para reproduzi-lo. Com tais características a Unity será IDE utilizada como base para o desenvolvimento do projeto.

Sobre a interface e utilização básica da Unity, ainda de acordo com o “Tutorial: Desenvolvimento de Jogos com Unity 3D” (PASSOS et al , 2007), “o motor de jogos Unity3D possui uma interface bastante simples e amigável que objetiva facilitar o desenvolvimento de jogos de diversos gêneros e outros sistemas de visualização. Sua área de trabalho é composta de várias janelas chamadas *views*, cada uma com um propósito específico”. Na Figura 1, é apresentada uma imagem com a interface da Unity, exemplificando os conceitos citados acima. Além disso, a interface pode ser modificada, possuindo estilos pré-definidos, dos quais o usuário pode escolher qual o mais agrada.



Figura 1: Interface do editor de cenas da Unity3D.

Fonte: Tutorial: Desenvolvimento de Jogos com Unity 3D, 2007.

2.3.1 Interface e utilização

Ainda utilizando das informações apontadas pelo “Tutorial: Desenvolvimento de Jogos com Unity 3D” (PASSOS et al , 2007), a interface e utilização básica da Unity pode ser dividida em: *Project*, *Hierarchy*, *Scene*, *Game* e a *Inspector*.

Project: “A janela *Project* é a interface para manipulação e organização do vários arquivos (Assets) que compõem um projeto tais como scripts, modelos, texturas, efeitos de áudio e Prefabs” (PASSOS et al , 2007);

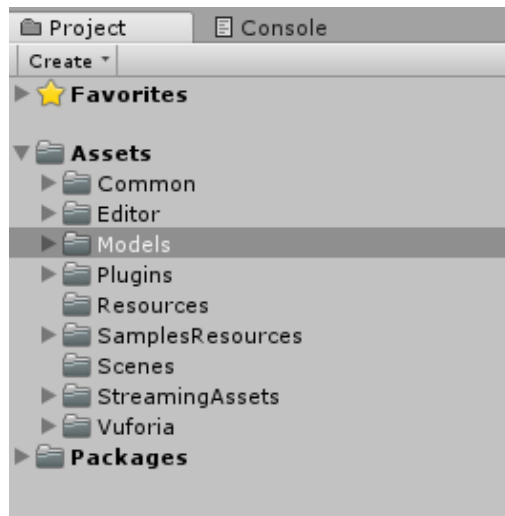


Figura 2: Interface da janela *Project*.

Fonte: Própria, 2018.

Hierarchy view: “A janela *Hierarchy* exhibe todos os elementos da cena que encontram-se na cena que se está editando. Além disso, nessa janela podemos organizar e visualizar a hierarquia de de composição entre os vários objetos que compõem a cena” (PASSOS et al , 2007);

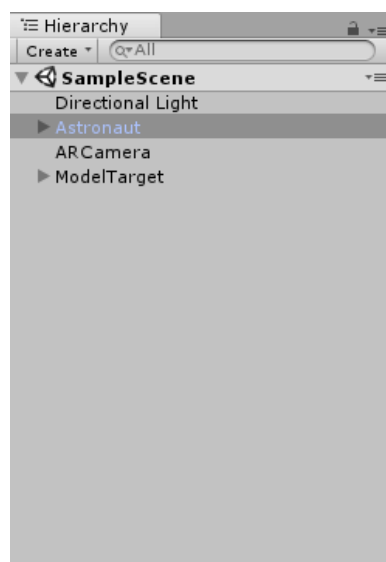


Figura 3: Interface da janela *Hierarchy*.

Fonte: Própria, 2018.

Scene: “A janela Scene é a forma principal de manipulação dos elementos visuais no editor de cenas da Unity, possibilitando a orientação e posicionamento desses elementos com um feedback imediato do efeito das alterações efetuadas. Nesta janela, pode-se manipular graficamente os objetos através das opções de arrastar e soltar com o mouse” (PASSOS et al , 2007);

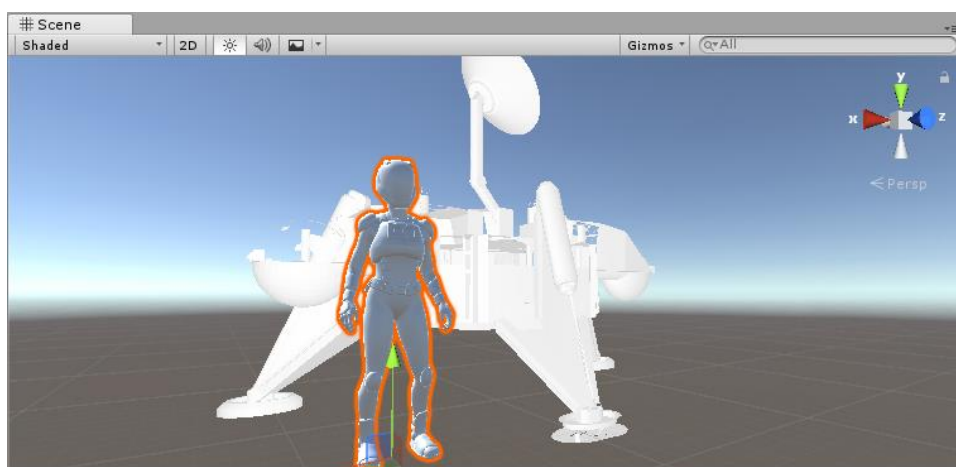


Figura 4: Interface da janela Scene.

Fonte: Própria, 2018.

Game: “A janela Game é responsável pela visualização da aplicação em desenvolvimento da forma que ela será exibida quando finalizada. Nessa janela, pode-se rapidamente ter uma prévia de como os elementos estão se comportando dentro da aplicação. Além disso, a Unity fornece a opção de se paralisar (botão pause) a simulação enquanto ela estiver em depuração, de forma a possibilitar que os parâmetros dos vários elementos possam ser ajustados para experimentação” (PASSOS et al , 2007);

Inspector: “Na janela Inspector, tem-se acesso aos vários parâmetros de um objeto presente no cenário, bem como aos atributos de seus componentes (Components)” (PASSOS et al , 2007).

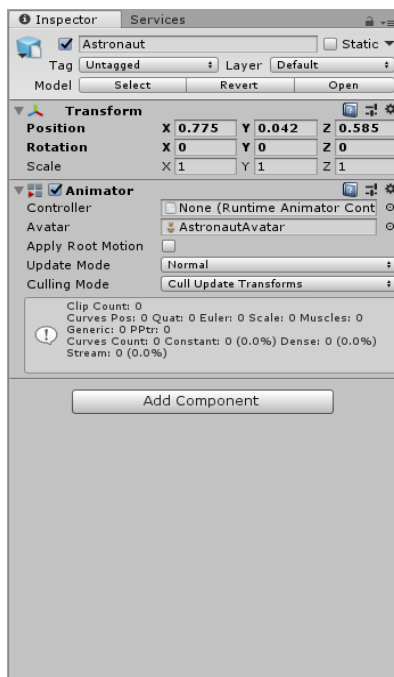


Figura 5: Interface da janela *Inspector*.

Fonte: Própria, 2018.

2.4 VUFORIA

O Vuforia é um kit de desenvolvimento de softwares (SDK), criado pela empresa Qualcomm, que permite a criação de aplicações móveis com realidade aumentada, se comunicando através de um plug-in da plataforma Unity, para maior interação e expansão das possibilidades de utilização deste kit, conforme assinalado no artigo “Estudo sobre a utilização de Vuforia e Unity 3D com RA para dispositivos móveis”(BERGAMASCHI.; DE MORAIS, 2014).

De acordo com a postagem sobre “*Pokémon GO Style Augmented Reality With Vuforia*” (Megali, 2016): “O Vuforia usa o *feed* da câmera do dispositivo, combinada com dados de acelerômetro e giroscópio, para examinar o mundo. O Vuforia usa a visão computacional para entender o que ‘vê’ na câmera e criar um modelo do ambiente”. Como assinalado nesta mesma postagem, existem múltiplas possibilidades para essa SDK desempenhar suas funções, as aqui

estudas serão: a utilização da localização global através de coordenadas, o reconhecimento de *Targets* (Alvos), reconhecimento de objetos simples e complexos, e, por último, o reconhecimento do terreno.

Assim, para desempenhar suas funções de reconhecimento de objetos o Vuforia se utiliza de *Targets* para colocar os objetos virtuais em nossa realidade (Library Vuforia, 2018). Conforme apresentado por Megali e a biblioteca de desenvolvimento do Vuforia, esses *Targets* são divididos nas seguintes funções e categorias:

- *Model Targets*(Modelos Alvo): estes alvos possibilitam ao Vuforia reconhecer modelos reais através de modelos 3D dos mesmos. Gerando interações com eles e um novo nível de interação com objetos reais, criando uma camada virtual sobre o objeto real. Como demonstrado na Figura 2;

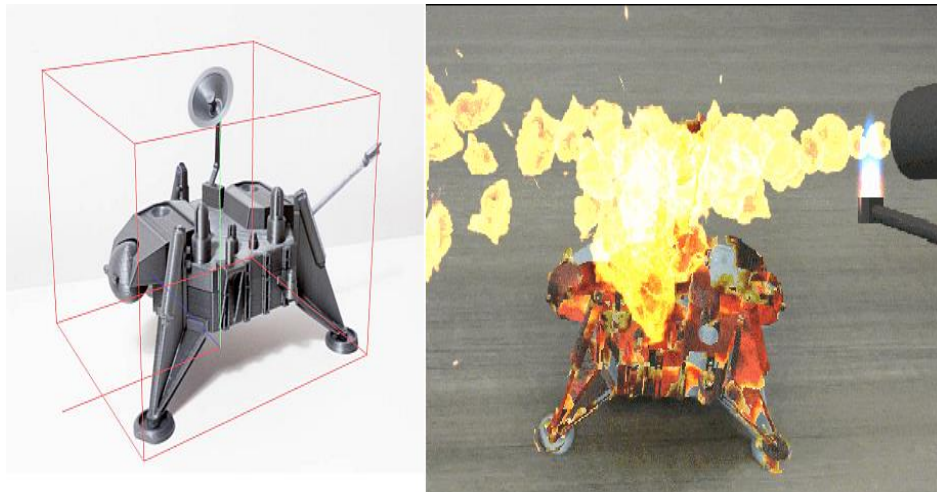


Figura 6: Interação de um Modelo *Target* e seu objeto real

Fonte: Library Vuforia, 2018.

- *Image Targets*(Imagens Alvo): o algoritmo do Vuforia pode identificar padrões de imagem escolhidos pelo desenvolvedor para serem alvos de posicionamento de objetos virtuais. Qualquer imagem pode ser usada para este fim, mas quanto mais detalhada e intrincada for a imagem,

maiores as chances do algoritmo reconhecê-la. Representado pela Figura 3;



Figura 7: Objeto projetado sobre um *Image Target*.

Fonte: Library Vuforia, 2018.

- *Multi-Targets*(Múltiplos-Alvos): são alvos com múltiplas superfícies que são *Image Targets* cada uma em um arranjo geométrico específico que pode ser identificado pelo Vuforia. A posição e orientação de cada *Image Target* em um objeto *Multi-Target* são relativas ao seu centro volumétrico;

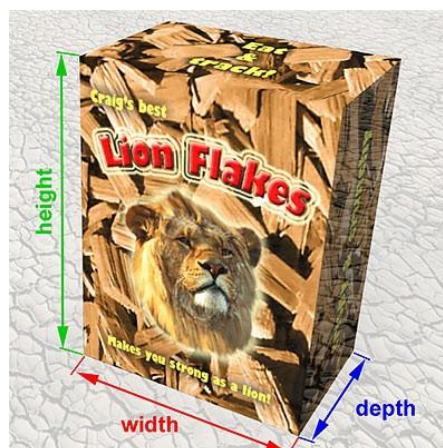


Figura 8: Objeto *Multi-Target* e suas dimensões.

Fonte: Library Vuforia, 2018.

- **Cylinder Targets:** alvos que consistem de objetos reais, normalmente embalagens, em forma cilíndrica ou cônica. Também suportam projeções de objetos em seu topo e em seu fundo de um destes objetos. O Vuforia detecta um objeto cilíndrico ou cone através de seu centro volumétrico, assim como um *Multi-Target*. A Figura 5 demonstra estas características;



Figura 9: Projeção de um *Cylinder Target* em um objeto real.

Fonte: Library Vuforia, 2018.

- **Object Target:** o Vuforia consegue detectar e rastrear objetos reais complexos ao realizar o escaneamento utilizando a ferramenta *Vuforia Object Scanner*, que gera um modelo 3D do objeto escaneado para manipulação do mesmo. Funciona melhor em objetos geometricamente estáveis, ou seja, que estejam em equilíbrio com seu plano geométrico, e que possuam uma superfície detalhada. Um exemplo desta função é apresentada na Figura 6;



Figura 10: Teste de um *Object Target* em um celular.

Fonte: Envato tuts+, 2018.

- *VuMarks*: são códigos de barras customizáveis, que são detectados pelo escâner do Vuforia, e podem armazenar informações como ids únicos, texto ou bytes. Funcionam similarmente a QRcodes;

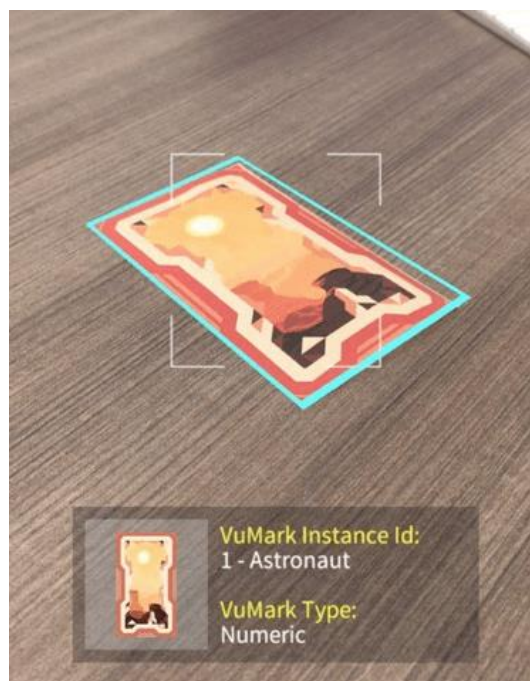


Figura 11: Um VuMark sendo escaneado e mostrando suas informações.

Fonte: Unity3D Blog, 2018.

- *Ground Plane* (Plano de Chão): dá a possibilidade de fixar objetos em superfícies horizontais em seu ambiente. Suportando a detecção e rastreamento de superfícies horizontais e também possibilitando a fixação de objetos através de *Anchor Points* (Pontos Ancora);



Figura 12: Demonstração da projeção de um objeto em Ground Plane.

Fonte: Vuforia Library, 2018.

A interpretação e projeção de objetos no terreno se dá através da ferramenta “*Smart Terrain*”, que conforme é informado pela biblioteca de desenvolvimento do Vuforia: “O *Smart Terrain* reconstrói, reconhece e rastreia objetos físicos e superfícies. Estes são representados como malhas 3D na cena Unity”. Em outras palavras, essa ferramenta usa a câmera do aparelho como um escâner, e assim que o mesmo reconhece o ambiente e seus objetos, a aplicação cria modelos 3D dos objetos reconhecidos, dando à aplicação a possibilidade de se organizar de acordo com o ambiente, de acordo com Megali (2016).

2.4.1 Vuforia e a Unity

Para realizar a ligação entre os objetos desenvolvidos para o Vuforia e aplicá-los a Unity é necessário registrar-se no site do Vuforia, e obter uma licença de API, indicado nas figuras x e x. Com esta licença em mãos, é necessário configurar sua instância do Vuforia em seu projeto na Unity.

Add License Key

Project Type

Select the type of project you are working on. See [Pricing](#) for more details.

- Development** - my app is in development
- Consumer** - my app will be published for use by consumers
- Enterprise** - my app will be distributed for use by employees

Project Details

App Name

 You can change this later

Device

- Mobile**
- Digital Eyewear**

License Key

- Develop - No Charge**

Figura 13: Tela de adição de chave de licença do Vuforia.

Fonte: Própria, 2018.

The screenshot shows the Vuforia Developer Portal interface. At the top, there is a green navigation bar with the Vuforia logo and the text 'vuforia™ Developer Portal'. Below this, there are several menu items: 'Home', 'Pricing', 'Downloads', 'Library', 'Develop' (which is highlighted), and 'Support'. Underneath the navigation bar, there are two sub-menu items: 'License Manager' and 'Target Manager'. The main content area shows the path 'License Manager > Nova Ventis Adven...'. Below this, the title 'Nova Ventis Adventurers' is displayed, followed by links for 'Edit Name' and 'Delete License Key'. There are two tabs: 'License Key' (selected) and 'Usage'. A message says 'Please copy the license key below into your app'. Below the message, a license key is shown: 'AQhd5tH/////AAABmSHsVglX/kUWvxVAXITybocTFTBss92tQVAg'. The license key is partially obscured by a red rectangular box.

Figura 14: Tela de visualização da chave de licença do Vuforia.

Fonte: Própria, 2018.

Dentro da interface da Unity, na *Inspector view*, temos acesso as configurações do Vuforia dentro do projeto, nela podese inserir a chave da API e assim ter acesso a base de dados de nosso projeto, onde estarão contidos os modelos e objetos de utilização do Vuforia.

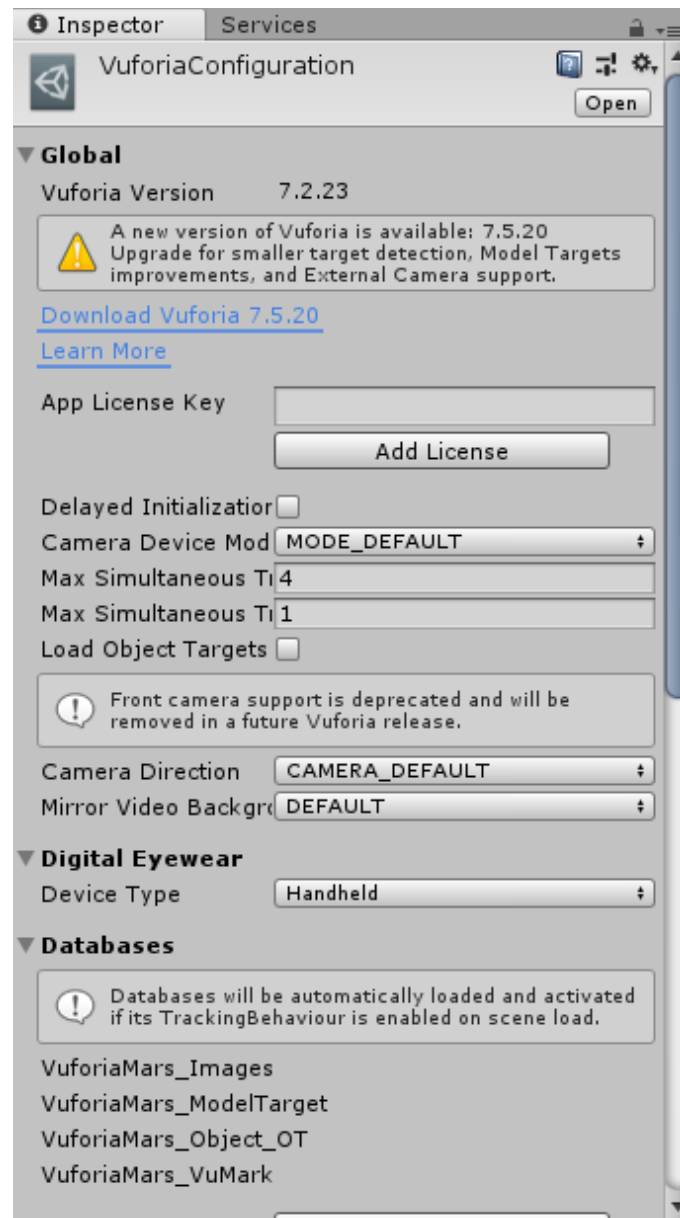


Figura 15: Configuração da instancia do Vuforia no projeto Unity.

Fonte: Própria, 2018.

2.5 GEOLOCALIZAÇÃO

A geolocalização é uma tecnologia relativamente nova, criada em 2009, que pode referenciar as coordenadas atuais de um aparelho que possua um software capaz de resgatar os dados necessários para a criação dessa informação. De acordo com o artigo “Geolocalização: conceitos e aplicações” (PALLADINO, 2018), sendo as maneiras de resgatar tais dados:

- GPS: a localização é realizada por satélite, captando no mínimo o sinal de três deles. Este processo pode demorar devido às condições do tempo;
- AGPS: chamado de GPS assistido é um aprimoramento do GPS. A tecnologia recebe dados de suporte (posição dos satélites) de uma antena de telefonia celular;
- GSM: a localização é feita por rastreamento através das ondas de rádio, utilizando as informações fornecidas pelas operadoras móveis, ou seja, é possível localizar um telefone celular mesmo com o GPS desativado, basta o dispositivo estar ligado e com sinal;
- Wifi: por meio das redes wifi disponíveis é possível determinar a geolocalização atual, onde o sinal GPS não está disponível.

A API escolhida para realizar a geolocalização deste projeto será a Mapbox, uma plataforma de serviços de geolocalização especializada em RA. “Mapbox AR fornece aos desenvolvedores 125 milhões de locais, informações comerciais de Foursquare e Factual, e dados de localização ao vivo agregados a partir de 300 milhões de usuários. Através do SDK do Google Maps para Unity, a plataforma também permite experiências multiusuários AR em que os usuários podem interagir em tempo real”. (PALLADINO, 2018)

A Mapbox disponibiliza um kit de desenvolvimento, que seria a “*Mapbox AR toolkit*”, e, conforme destacado acima, pode ser usada na plataforma de desenvolvimento Unity.

2.6 BLENDER

“A ferramenta Blender foi concebida em dezembro de 1993 e tornou-se um produto oficialmente em agosto de 1994 como uma aplicação integrada que permite a criação de uma diversificada gama de conteúdos 2D e 3D” (OLIVEIRA, 2018), sendo assim, o Blender é uma ferramenta de modelagem e animação versátil, que pode atender todos os nichos gráficos necessários para o desenvolvimento desse projeto.

Para a criação dos objetos, o Blender possui formas básicas, materiais, tipos e até mesmo comportamentos, que são aplicados ao objeto que será modelado e/ou animado. “Através de transformações que deformam o objeto ou a forma primitiva, é possível conseguir grandes resultados” (OLIVEIRA, 2018, p.5), como indicado na Figura 12.

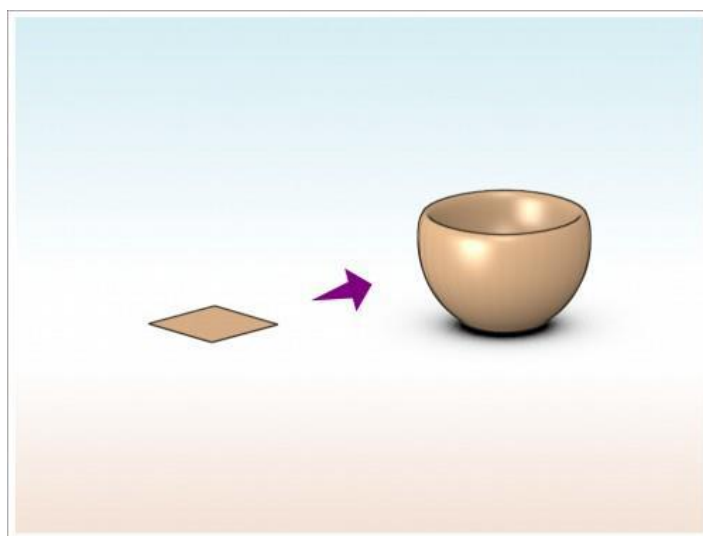


Figura 16: Criação de um pote a partir da deformação de um objeto.

Fonte: Introdução a Modelagem 3D com Blender 3D, 2013.

2.7 GAMIFICAÇÃO

Gamificação é "o uso de elementos de design de jogos em contextos fora de jogos para tornar sistemas ou processos mais divertidos e envolventes" (AMIR; RALPH, 2017) o uso destes elementos vai além da engenharia de software ou design de uma aplicação e suas funções, necessitando de mecânicas que incentivem a interação do usuário e que atraiam sua atenção para a ação, processo ou sistema que se deseja gamificar e também é necessário que se encaixem mutuamente, apresentando um *feedback* conciso, satisfatório e dinâmico ao usuário.

Para isso, possuímos fatores que afetam a efetividade da gamificação, segundo o artigo "Propondo uma Teoria da Eficácia da Gamificação", são esses:

- A motivação intrínseca: "depende dos sentimentos de autonomia, competência e correlatividade, causados pelas mecânicas empregadas no jogo, respectivamente" (AMIR; RALPH, 2017), através da liberdade de escolha, desafios e encorajamento a comunicação e relacionamento entre os jogadores;
- A motivação extrínseca: "os jogos digitais empregam inúmeros motivadores extrínsecos incluindo pontos, níveis, crachás ou troféus" (AMIR; RALPH, 2017). Assim, podemos fazer uma ligação de motivações extrínsecas dos jogos com as do mundo real, entregando recompensas virtuais pelas ações reais tomadas e/ou de acordo com metas batidas;
- Mecânicas de Jogo: "Mecânica do jogo, incluindo espaço, objetos, ações, regras e habilidades, são o núcleo de um jogo" (AMIR; RALPH,

2017). Todas as mecânicas citadas são manipuladas e usadas, seja virtualmente, seja no plano físico;

- Dinâmicas Imersivas: “As dinâmicas imersivas são fatores que afetam a imersão do jogador no sistema ou atividade gamificada” (AMIR; RALPH, 2017), normalmente englobam a forma de apresentação da atividade gamificada, tanto visualmente, quanto narrativamente.

Gamificar uma atividade ou processo não consiste apenas em apresentar pontuações, rankings ou recompensas ao usuário, mas sim em utilizar toda uma lógica de regras, objetivos e interatividade com o objeto de gamificação para aumentar o engajamento e incentivar o usuário.

Deve-se ressaltar a necessidade do estudo sobre as atividades a serem gamificadas, os impactos que tais atividades podem causar e as formas de aplicar a gamificação a essas atividades.

3 PROPOSTA DE TRABALHO

Este projeto propõe-se então ao estudo e planejamento do desenvolvimento de um jogo, utilizando da Unity para a programação das funções, mecânicas e ações do jogo, junto do Blender para a criação dos objetos, *sprites*, personagens e animações necessárias. Para que assim, após a conclusão da fase de modelagem e criação dos objetos e seus comportamentos necessários, implementar as interações com RA através do Vuforia, junto do Mapbox, para determinar onde os eventos e objetos aparecerão no mundo real, e como os jogadores poderão interagir com eles.

E através de pesquisas, propor aplicações de gamificação para amplificação do interesse e engajamento do jogador em certas atividades e, além disso, incentivo para o seu desenvolvimento pessoal e de suas relações sociais.

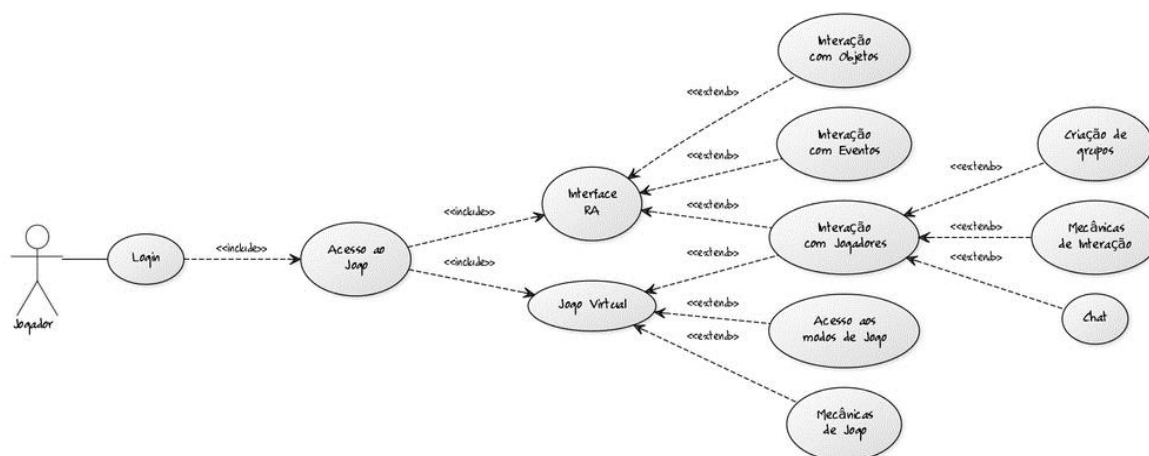


Figura 17: Diagrama de representação do funcionamento básico.

Fonte: Própria, 2018.

4 APLICANDO A RA

Para exemplificar e demonstrar as aplicações do Vuforia junto da Unity para a aplicação da RA, será usado um projeto de exemplo, com base no projeto desenvolvido e disponibilizado pelos “*The Knights of Unity*” (2017), em seu blog.

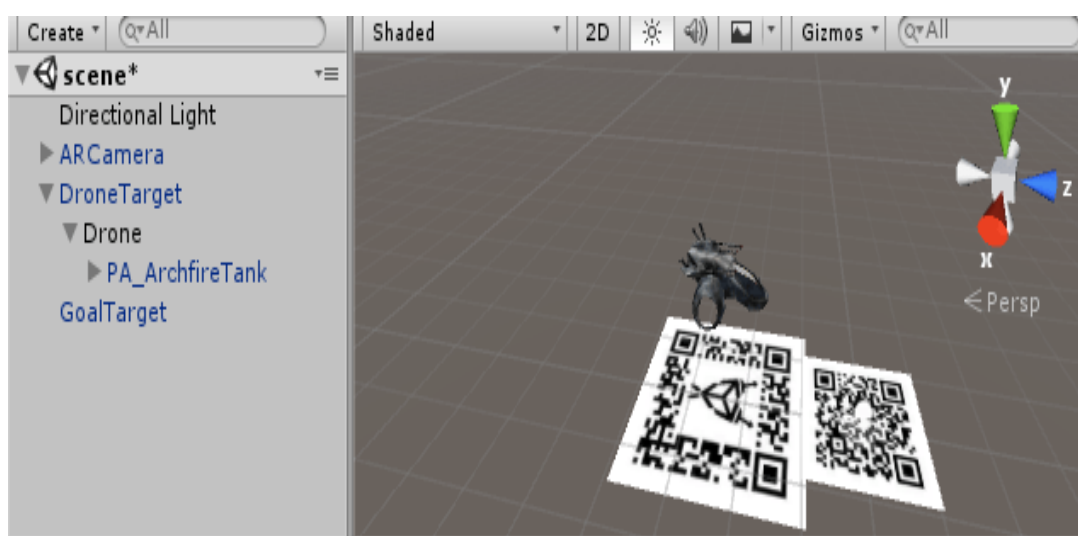


Figura 18: Demonstração da aplicação exemplo.

Fonte: The Knights of Unity, 2017.

O objetivo dessa aplicação é fazer com que o objeto *Drone* apareça sobre o *DroneTarget* e desloque até a marca no objeto *GoalTarget*, sendo que estes marcadores serão objetos reais, com os códigos de barra impressos sobre eles. Para isso, serão necessárias as configurações dos objetos apresentados e suas propriedades e comportamentos. Todos os objetos 3D, texturas e marcadores foram disponibilizados pelos The Knights of Unity, em sua postagem Unity AR Tutorial: Augmented Reality Game Development with Vuforia (The Knights of Unity, 2017),

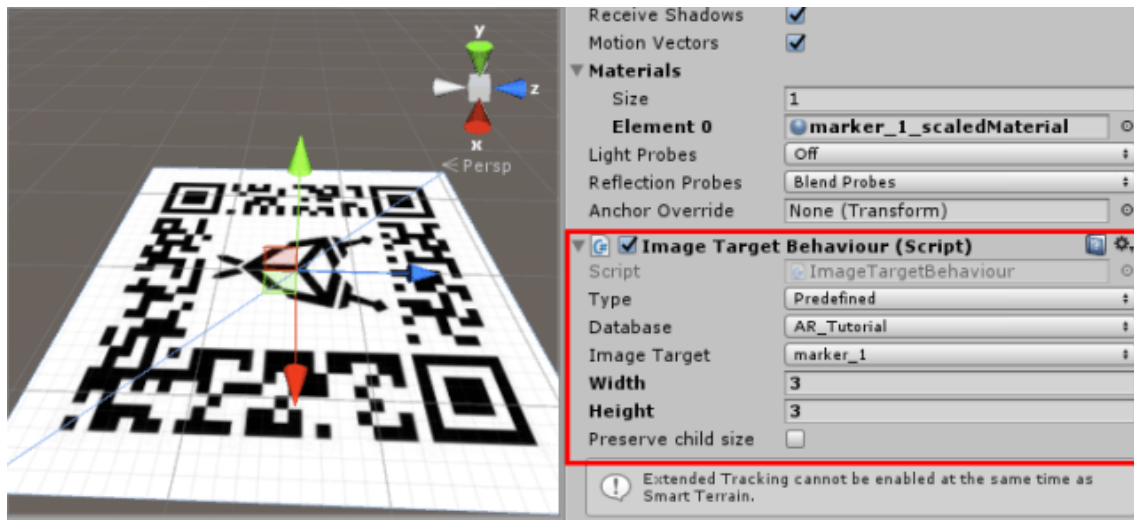


Figura 19: Configuração do *DroneTarget*.

Fonte: The Knights of Unity, 2017.

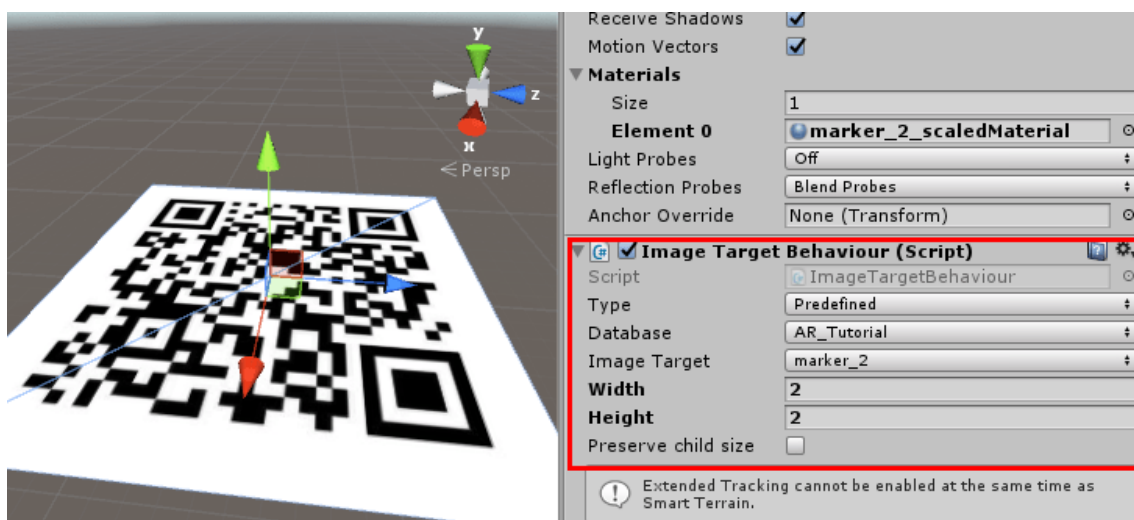


Figura 20: Configuração do *GoalTarget*.

Fonte: The Knights of Unity, 2017.

Já o objeto do *Drone* será um objeto modelo, inserido dentro de um objeto chamado *Drone* que está sob o *Drone Target* na hierarquia do Unity, como pode ser visualizado na Figura 17.

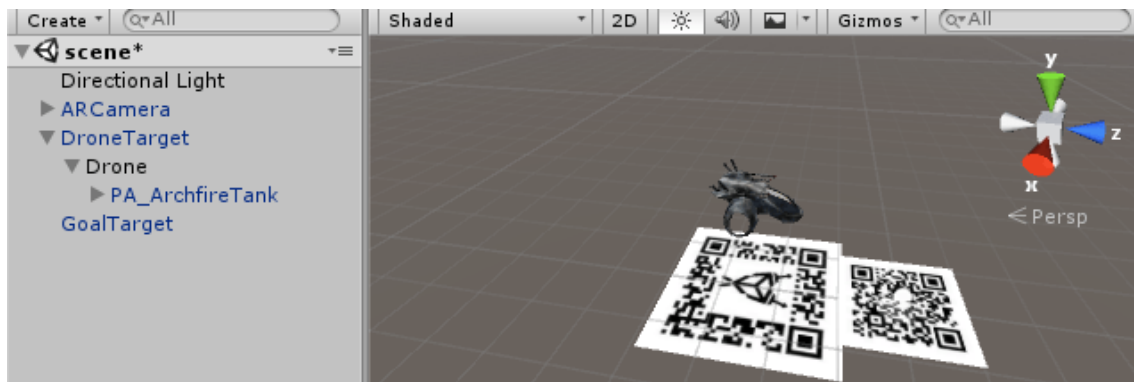


Figura 21: Configuração do objeto *Drone*.

Fonte: The Knights of Unity, 2017.

Utilizando da interface é possível, como botão direito do *mouse*, adicionar um objeto modelo novo sob o objeto *Drone*, também inserindo as configurações de escala e posicionamento do objeto modelo em relação ao objeto marcador, como é sinalizado pelo post Unity AR Tutorial: Augmented Reality Game Development with Vuforia (The Knights of Unity, 2017), onde o *Drone* possui um objeto abaixo dele em sua hierarquia, o qual é o próprio modelo 3D de um *Drone* customizado.

Após isso, é necessário adicionar um componente de configuração ao *Drone*, o *NavMesh* que será responsável pela movimentação do mesmo e um objeto ao *GoalTarget* para identificação e destaque dele.

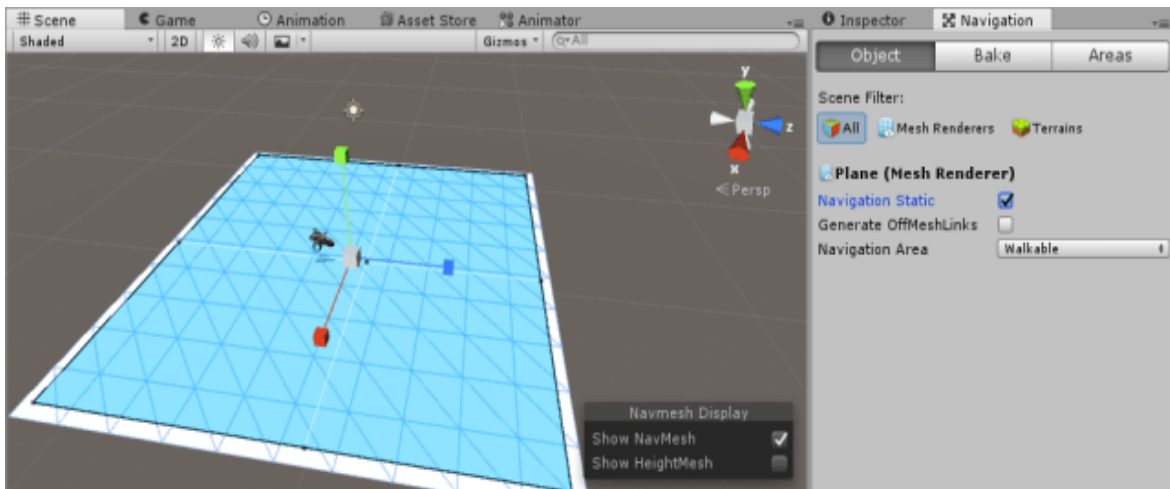


Figura 22: Visualização do *NavMesh*.

Fonte: The Knights of Unity, 2017.

Após criar o *NavMesh*, para que o objeto não atrapalhe a visualização e manipulação da cena e dos outros objetos, é necessário marcar a opção *Navigation Static* e pressionar o botão *Bake*, assim o *NavMesh* pode ser deletado e ainda permanecer no projeto. Com isso, é preciso então adicionar o *NavMesh* a nosso objeto *Drone*.

Já o objeto de destaque que residirá dentro do *GoalTarget*, será um objeto em forma de anel. Que será criado utilizando *sprites* já disponíveis na Unity, e apenas alterando sua cor e tamanho.

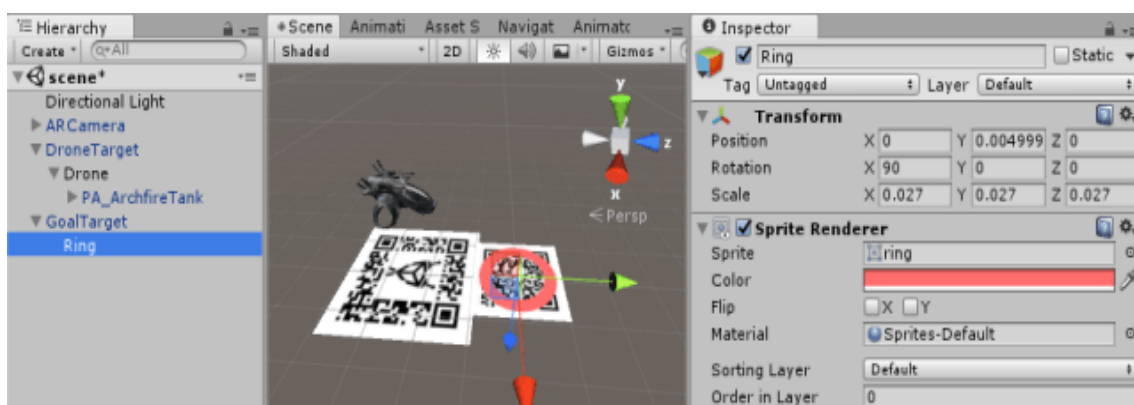


Figura 23: Criação do objeto *Ring*.

Fonte: The Knights of Unity, 2017.

Como é indicado no post Unity AR Tutorial: Augmented Reality Game Development with Vuforia (The Knights of Unity, 2017), onde é necessário apenas declarar as variáveis de controle dos objetos, com sua tipagem respectiva que referencia aos métodos do objeto ao qual é relacionada, e uma variável do tipo *Vector3*, que controla posições em três dimensões.

```
//destination marker reference
public GameObject markerGoal;
//parent position
Vector3 parentPos;
//navmesh component reference
NavMeshAgent agent;
```

Figura 24: Declaração das variáveis de referencia dos objetos.

Fonte: The Knights of Unity, 2017.

Então é necessário inicializar a variável de controle do objeto de movimento do *Drone*, no caso a variável *agent*, que receberá o componente.

```
void OnStart()
{
    agent = GetComponent<NavMeshAgent>();
}
```

Figura 25: Inicialização da variável *agent*.

Fonte: The Knights of Unity, 2017.

E então criar uma função para este objeto, que se encarregará do rastreamento do *Ring* dentro do *GoalTarget* e movimentará o *Drone* até ele, a qual deve estar dentro da função *Update*, que é executada a cada *frame* que se passa.

```

void MoveToTarget()
{
    if (markerGoal.active)
    {
        parentPos = markerGoal.transform.parent.position;
        agent.SetDestination(parentPos);
    }
}

```

Figura 26: Função de movimentação do *Drone*.

Fonte: The Knights of Unity, 2017.

Assim, deve-se adicionar o *Ring* como uma variável pública em seu inspetor e então selecioná-lo como objeto da função dentro do *GoalTarget*.



Figura 27: Configuração do scrpt de movimentação.

Fonte: The Knights of Unity, 2017.

Nesta etapa do projeto o *Drone* já consegue se mover até o objeto *Ring* dentro do *GoalTarget*, mas ainda faltam alguns detalhes. Como a sombra do *Drone* e algo para enfatizar mais sua *movimentação*.

Para enfatizar a movimentação do Drone, é possível fazer o objeto se inclinar em sua movimentação até o marcador objetivo. Para isso será utilizada uma função chamada *MapRange*, que como é indicado pelos The Knights of Unity na publicação *Unity AR Tutorial: Augmented Reality Game Development with Vuforia* (2017), consiste em converter números baseados no alcance de um objeto para o alcance de outro objeto, um exemplo de aplicação utilizada neste mesmo post é a distância entre dois objetos, quando um objeto se aproxima do seu objetivo ou se afasta dele, podemos utilizar está função para realizar mudanças em cor, movimentação e outras características de um ou ambos os objetos.

```

//s - observed value
//a1 - min s value, a2 - max s value
//b1 - min result value, max result value
float MapRange(float s, float a1, float a2, float b1, float b2)
{
    if (s >= a2) return b2;
    if (s <= a1) return b1;
    return b1 + (s - a1) * (b2 - b1) / (a2 - a1);
}

```

Figura 28: Código da função *MapRange*.

Fonte: The Knights of Unity, 2017.

Com a função pronta, basta adicioná-la a uma função de controle do *Drone* que será chamada dentro da função *Update*, e então referenciá-la no objeto.

```

void PitchCtrl()
{
    transform.GetChild(0).eulerAngles = new Vector3(
        MapRange(agent.velocity.magnitude, 0f, 2f, 0f, 20f),
        transform.eulerAngles.y,
        transform.eulerAngles.z
    );
}

```

Figura 29: Função de controle da inclinação do *Drone*.

Fonte: The Knights of Unity, 2017.

A sombra do *Drone* será um *sprite*, que usará uma textura disponibilizada pelos The Knights of Unity na mesma postagem.

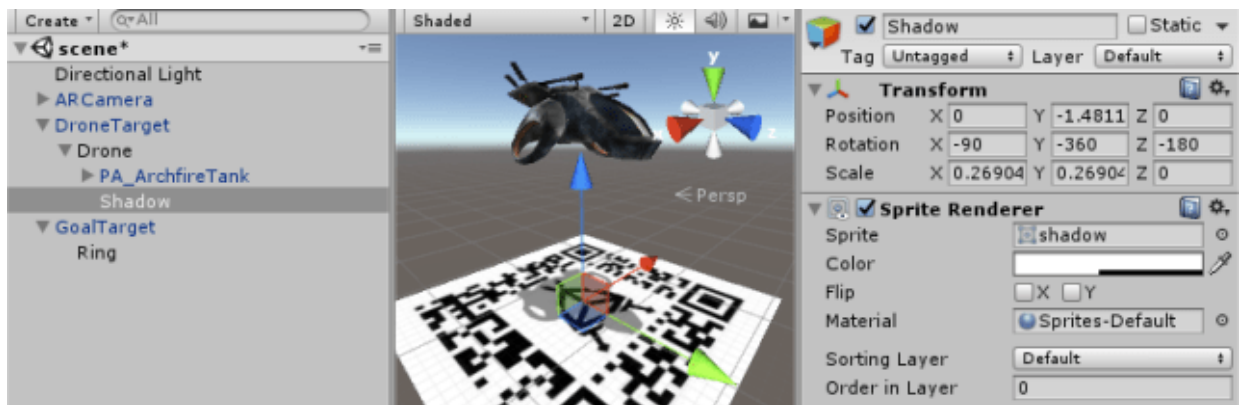


Figura 30: Configuração da sombra do *Drone*.

Fonte: The Knights of Unity, 2017.

Seguindo com o projeto, pode-se destacar ainda mais o marcador do objetivo adicionando uma linha, que servirá como guia caso o marcador se eleve acima do *Drone*, e um objeto *Destination* na base desta linha, o que dará uma visualização mais amigável e limpa do objetivo.

A linha será criada a partir de um material inserido em um *sprite* quadrado, chamado *Laser*, que terá seu *shader* alterado para o *VertexLit Blend* e configurando as partículas do *sprite* para as exibidas na Figura 31, a frente do *Tiling*.

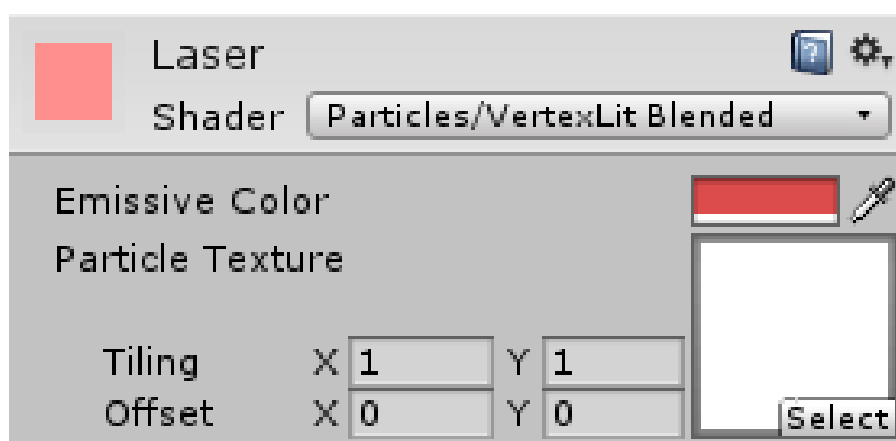


Figura 31: Configurando a linha.

Fonte: The Knights of Unity, 2017.

Após a criação do material, deve-se adicionar um *Line Renderer* ao objeto *Ring*, utilizando como material e criar um *scprit*, chamado *Goal*, que será ligado ao mesmo, que tem como função manter a largura da linha em um tamanho determinado e reter sua posição no centro do objeto *Ring*.

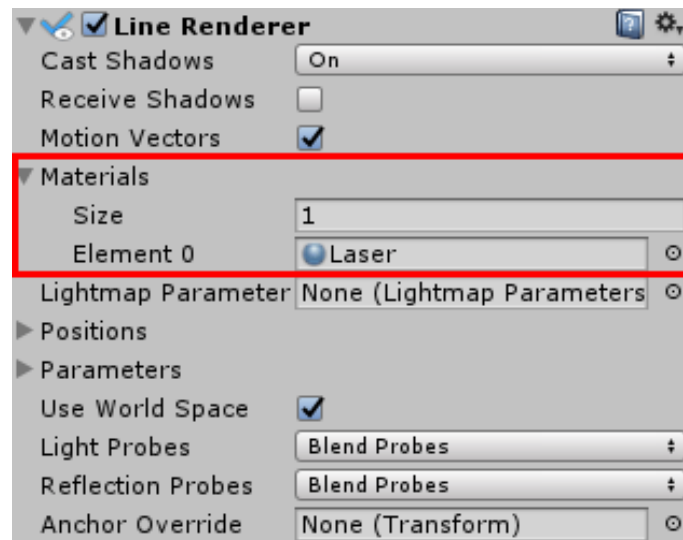


Figura 32: Configuração do Line Render dentro do objeto Ring.

Fonte: The Knights of Unity, 2017.

```
public class Goal : MonoBehaviour {

    LineRenderer line;

    void Start () {
        line = GetComponent<LineRenderer>();
        line.SetWidth(.05f, .05f);
    }

    void Update () {
        //current ring position
        line.SetPosition(0, transform.parent.position);
        //respective ring position on the ground
        line.SetPosition(1, new Vector3(transform.parent.position.x, 0, transform.parent.position.z));
    }
}
```

Figura 33: Código da função *Goal*.

Fonte: The Knights of Unity, 2017.

O objeto *Destination* será criado com um *Sprite Renderer* que terá como forma um círculo.

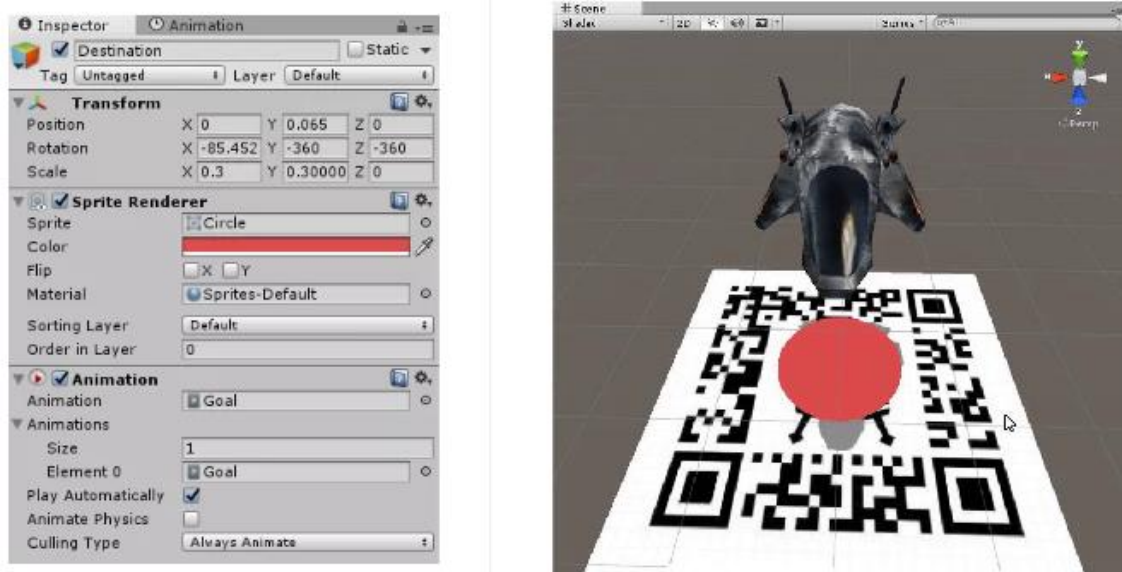


Figura 34: Criação e visualização do objeto *Destination*.

Fonte: The Knights of Unity, 2017.

E por fim, basta apenas modificar a função *MoveToTarget*, adicionando uma variável pública *destSprite*, que conterà o objeto *Destination*, e o fará acompanhar o marcador do *GoalTarget*, assim o projeto está finalizado.

```
void MoveToTarget()
{
    if (markerGoal.active)
    {
        parentPos = markerGoal.transform.parent.position;
        agent.SetDestination(parentPos);
        destSprite.transform.position = new Vector3(parentPos.x, 0, parentPos.z);
    }
}
```

Figura 35: Modificação na função de movimentação.

Fonte: The Knights of Unity, 2017.

5 GAMIFICANDO

Antes de abordar a estética, história, regras, objetivos, mecânicas de gameplay e funcionalidades do jogo, deve-se pontuar o núcleo da gamificação, as mecânicas que levarão a sua aplicação e como funcionam, sendo elas o cerne da gamificação e constituem a maior parte do sucesso da aplicação da mesma.

5.1 MECÂNICAS

Neste projeto, busca-se ampliar o interesse, produtividade, engajamento dos jogadores em seu autodesenvolvimento em geral, além de procurar promover mudanças de comportamento e incentivo do desenvolvimento de comportamentos e práticas produtivas e/ou positivas em sua vida.

Tais mecânicas buscam se mesclar com as mecânicas nativas do gameplay para levar a maior imersividade do jogador, sendo completamente necessárias para o sucesso do mesmo em seu avanço no jogo, gerando um paralelo com sua vida. Elas se darão através:

- Das informações que o próprio jogador insere no jogo, por meio de posts na comunidade ou pessoais, que neles o usuário informará qual atividade desenvolvida e por quanto tempo foi desenvolvida;
- De atividades e objetivos pré-existent no próprio jogo, que ao serem realizadas e informadas ao mesmo, recompensarão o jogador;
- De uma comunidade e de interações entre os jogadores e suas atividades nela;
- De mecânicas de incentivo a interação social, que buscam unir os jogadores e a comunidade.

Assim, o jogo irá gerar as recompensas e feedbacks respectivos ao jogador.

No resto deste capítulo serão apresentados elementos da gamificação mais a fundo e como se dá o seu funcionamento.

5.2 ATRIBUTOS IN-LIFE

Os atributos são representações das habilidades do personagem em seu jogador, as habilidades físicas são representadas pelos atributos: força, destreza, agilidade e constituição. As habilidades psicológicas, sociais e cognitivas são representadas por: inteligência, sabedoria, carisma e criatividade.

As atividades realizadas pelo jogador, que são informadas ao aplicativo, lhe recompensarão com “pontos de experiência” nos atributos relacionados a elas. Em determinadas quantidades, os “pontos de experiência” aumentarão o total do atributo ao qual pertencem, dando ao jogador ciência do seu progresso e o recompensando pela continuidade de seu desenvolvimento.

As atividades também podem levar a interação entre outras atividades e entre atributos, de acordo com certos critérios, dando uma maior noção ao jogador no que elas o influenciam e como o fazem.

Vale-se ressaltar que as atividades não costumam afetar apenas um atributo, normalmente afetam uma variedade deles.

5.2.1 Força

A “Força” representa a força física do jogador, sendo adquiridas através de exercícios e atividades físicas que levem a utilização de seus músculos. O aplicativo poderá procurar a incentivar o jogador a realizar certos tipos de exercícios, baseados na preferência do mesmo, utilizando seu personagem como base. Com um maior nível de pesquisa e interação com o usuário, pode-se utilizar o biótipo do mesmo como base para a recomendação de atividades físicas.

5.2.2 Destreza

“Destreza” é a finesse, aptidão manual, habilidade movimentação, facilidade e amplitude de movimento do jogador, sendo afetada por atividades que

procuram melhorar as capacidades motoras e manuais, incluindo a flexibilidade e movimentação. Podendo se relacionar com atividades físicas de alongamento, equilíbrio, controle e sincronia de movimento, assim como habilidades de escrita, pintura, desenho e artesanato do jogador.

5.2.3 Agilidade

“Agilidade” trata-se da velocidade de movimentação, reflexos, a capacidade de movimentação e pensamento rápido do jogador, sendo afetada por atividades físicas, manuais e cognitivas que causem aprimoramento dessas capacidades. Pode se relacionar com atividades físicas de corrida, equilíbrio e funcionais, já no âmbito cognitivo costuma a se atrelar com estímulos de reflexo e rapidez de pensamento, como em jogos.

5.2.4 Constituição

Este atributo representa a saúde e constituição física do jogador, influenciando sua capacidade física em sua totalidade e sua qualidade de vida. Sendo afetada principalmente por uma boa alimentação e prática de exercícios físicos que buscam a melhoria da resistência do praticante.

5.2.5 Inteligência

A “Inteligência” é a capacidade do indivíduo de memorizar, pensar e resolver problemas lógicos, sendo um atributo que será principalmente afetado por leitura, estudo e prática de exercícios cognitivos e lógicos. Além de se relacionar com todas as habilidades que podem envolver a memória e pensamento rápido e lógico do jogador.

5.2.6 Sabedoria

Já a “Sabedoria” é responsável por representar a percepção, bom senso, intuição, sentidos e conhecimento próprio e do mundo que o cerca, que o jogador possui. Este atributo é afetado principalmente por atividades que elevem a reflexão do jogador, seu senso crítico e que ampliem suas perspectivas, além de, no âmbito físico, as que melhorem seus sentidos e sua forma de utiliza-los.

5.2.7 Carisma

O atributo “Carisma” engloba a força de personalidade, magnetismo pessoal, habilidades de relacionamento social, comunicação e a simpatia do jogador. Se manifestando, e sendo desenvolvido, com atividades que melhorem as habilidades sociais do jogador, lidando com pessoas, relacionamentos e ideias.

5.2.8 Criatividade

A “Criatividade” é um conceito que procura incorporar a capacidade de aprender, criar e absorver informações e experiências do jogador. Influenciando e sendo influenciado pelos outros atributos psicológicos e cognitivos do indivíduo.

5.2.9 Relacionamento entre os atributos

Como dito anteriormente, os atributos podem se relacionar e gerar interações entre si, assim manter todos seus atributos em níveis “saudáveis” pode recompensar ainda mais o jogador. Por exemplo, manter uma boa constituição levará a maior facilidade de adquirir experiência em atividades relacionadas aos outros atributos físicos do jogador.

5.3 ATIVIDADES

As atividades serão informadas pelo jogador, através do aplicativo, junto do tipo de atividade, dificuldade e por quanto tempo foi prática, juntamente com a localização atual do usuário. Assim o aplicativo irá avaliar a atividade e a quantidade de tempo utilizada, gerando uma quantidade de experiência para as habilidades envolvidas com essa atividade.

Existem também atividades criadas previamente no aplicativo, que poderão ser completadas pelo jogador, cada uma afetando um conjunto de habilidades e propondo um objetivo claro. Entre elas existirão as atividades diárias, únicas, “repetitivas” e missões. Cada uma terá seu próprio nível de bonificações ao jogador e dificuldade.

- **Atividades Diárias:** são atividades que serão dadas ao jogador diariamente que podem ser completadas uma vez ao dia, normalmente repetindo-se, mas seus níveis de dificuldade e quantidade serão baseadas no nível dos atributos do jogador e seu personagem. Jogadores mais experientes terão acesso a mais e mais complexas atividades diárias, enquanto os jogadores novatos terão de melhorar suas pontuações para terem acesso a mais dessas atividades;
- **Atividades Repetitivas:** são compostas de tarefas mais simples que as atividades diárias, que podem ser realizadas mais de uma vez por dia. Seguindo as mesmas regras de limitações e complexidade das atividades diárias;
- **Missões:** são desafios propostos aos jogadores quando completam certos requerimentos, baseados em seus atributos, como tem desenvolvido seu personagem e em seu andamento dentro do jogo, que tendem a ser difíceis, mas recompensando o jogador muito mais que atividades repetitivas ou diárias.
- **Atividades únicas:** são objetivos maiores, que envolvem grande desenvolvimento do jogador e de seu personagem, levando muito tempo para serem desbloqueadas, mas trazendo recompensas únicas a seus personagens.

5.4 GRUPOS E A COMUNIDADE

Visando o auxílio e a melhora da interação social entre os jogadores, o jogo terá uma mecânica de grupos e possuirá uma comunidade onde os jogadores poderão interagir entre si. Propondo recompensas e incentivos para a interação entre os jogadores.

5.4.1 GRUPOS

Os grupos são criados quando jogadores, que se encontrem ou possuam um ao outro em sua lista de amigos, convidam-se para formar um grupo. O grupo é

uma parte essencial para as “Aventuras” (serão explicadas no capítulo das mecânicas *in-game*), além de recompensar jogadores que joguem em grupos, já que as recompensas de, por exemplo, derrotar um monstro não serão divididas, mas serão geradas unicamente para cada jogador. Assim os jogadores têm mais chances de receberem itens que procuram jogando em grupo. Além de, também, devido a seus números podem enfrentar monstros mais fortes que seus personagens sozinhos, podendo perseguir desafios maiores e maiores recompensas junto de seus companheiros.

5.4.2 COMUNIDADE

Consiste de uma comunidade online, conectada ao jogo, onde os jogadores podem compartilhar suas experiências dentro do jogo e fora dele, mostrando suas atividades, interagindo com outros jogadores e até mesmo ajudando-os.

Sendo dividida entre a comunidade social (ou “Plaza”) e o fórum, a comunidade social tem como objetivo recompensar os jogadores pela interação com outros e proporcionar mais uma forma dos jogadores se conhecerem e formarem grupos. Já o fórum, busca ser uma base de conhecimentos de jogo e auxílio aos jogadores na busca de se melhorar, consequentemente melhorando seus atributos, lá os jogadores podem divulgar seus conhecimentos e experiências. Além de que estas interações também cumprirem requisitos de missões dentro do jogo.

5.4.2.1 Plaza

No “Plaza” os usuários poderão fazer avaliações dos *posts* e comentários nos próprios ou nos *posts* de outros, estas interações geram recompensas aos responsáveis pelo *post* ou comentário, as avaliações possíveis são:

- Curtir: para cada nível de “curti” que a postagem ou comentário receber, o criador receberá um bônus de experiência relacionada ao atributo que a postagem foi atrelada;

- Prestativo: para cada nível de “prestativo” o usuário que realizou a postagem ou comentário recebe um bônus de experiência geral para o personagem.

Os comentários também serão usados como método de avaliação, quanto mais comentários maior a visibilidade do post na comunidade.

Mas também, como forma de prevenir comportamentos negativos e improdutivos, os jogadores também podem reportar um comentário ou post. Baseado na quantidade de reportagens, o post ou comentário serão apagados e junto dele qualquer recompensa ou bônus provindos do mesmo, e o jogador pode ser suspenso ou banido da comunidade e/ou jogo. Isso também dará uma forma de controle maior e maior visibilidade para jogadores que trapaceiem.

Além de que, os jogadores podem também enviar requisições de amizades para os criadores de posts ou comentários que visualize na comunidade.

5.4.2.2 Fórum

O Fórum é uma comunidade diferente, onde os jogadores criam discussões e posts, onde procuram ajudar outros ou expor seu conhecimento e experiências em seu desenvolvimento, estes são pelos atributos e categorias relacionadas a estes atributos.

Os posts do fórum são avaliados através da quantidade de respostas em um post ou discussão e marcações de positivo ou negativo, a quantidade de respostas aumenta a posição no ranking da categoria do post ou discussão, já as marcações de positivo ou negativo, reverberam como bônus no atributo relacionado à categoria do post ou reduções dos bônus do mesmo, respectivamente. Assim como nos posts do Plaza, jogadores podem reportar posts do Fórum.

Devido ao maior nível de complexidade os posts do Fórum podem acarretar a recompensas ainda maiores do que as recebidas no Plaza.

6 O JOGO

Neste capítulo serão apresentados todos os detalhes do jogo, como funcionaram as mecânicas *in-game*, a história, *design* e *gameplay*. Além de como esses elementos dentro do jogo podem interagir com as mecânicas de gamificação e como o jogador.

Foram utilizados conceitos, técnicas e mecânicas baseadas fortemente no manual de jogo Tormenta PRG.

6.1 HISTÓRIA

A história se baseia na vinda dos personagens dos jogadores de uma outra realidade, precisamente do continente de Nova Ventis que existe em tal realidade, o qual está sendo destruído por um grande cataclisma. Assim para escapar os reinados de Nova Ventis se uniram e num esforço coletivo criaram um portal por onde seus habitantes poderiam escapar assim os habitantes dos reinados usam desse portal para chegar a nossa realidade. Mas eles não vieram sozinhos, monstros e partes do cataclisma que afetava seu mundo vieram para o nosso. Assim os habitantes de Nova Ventis se juntam aos jogadores para combater este mal que agora se espalha também por nosso mundo, em forma de monstros e locais “amaldiçoados” onde os monstros e a corrupção que levou a destruição de seu mundo nascem.

6.2 COMEÇANDO A JOGAR

Para que o jogador possa começar a jogar, é necessário possuir um aparelho com sistema operacional Android, que possua câmera, acesso a internet, sistema de GPS e capacidade de sistema para suportar o jogo e seus

requisitos. Tendo isso, é necessário se registrar utilizando seu email ou sua conta google. Após ter criado sua conta e ter entrado no jogo o jogador deverá criar seu personagem.

6.3 OS PERSONAGENS

Os personagens dos jogadores são criados em passos, sendo uma das partes mais importantes para a imersividade e conexão do jogador com seu personagem.

- **1º Passo:** o jogador deverá escolher a raça de seu personagem, podendo customizar de suas características físicas e dar-lhe um nome;
- **2º Passo:** depois de escolher estas características de seu personagem, o jogador escolherá a especialização de seu personagem;
- **3º Passo:** após isso deverá decidir o valor de seus atributos iniciais do personagem, seguindo a mesma lógica dos atributos reais do jogador, mas com efeitos adaptados para as mecânicas dentro de jogo;
- **4º Passo:** após decidir seus valores de atributos o jogador deverá decidir suas habilidades iniciais, para determinar quais são as capacidades e funções de seu personagem;
- **5º Passo:** então se deve escolher quais serão os talentos iniciais do personagem;
- **6º Passo:** com seus talentos decididos, chega a hora de escolher as perícias do personagem;

Assim, o personagem do jogador estará pronto e poderá ser criado, o jogador pode a qualquer momento retornar os passos da criação e modificar seu personagem a sua vontade.

O personagem é a parte crucial para a implementação das técnicas de gamificação apresentadas, sendo ele a ligação entre o jogador e as mecânicas de desafios e recompensas apresentadas neste projeto.

6.3.1 Atributos

Os atributos dentro de jogo funcionam de forma similar do que a explicada quando são relacionados às habilidades reais do jogador, mas adaptadas para as mecânicas de jogo. As premissas das habilidades são as mesmas, então neste tópico serão abordados seus efeitos dentro de jogo, assim como as funções que englobam.

Os atributos são quantificados em “Valor” e “Modificador”, os valores são a quantidade de pontos que o jogador possui em seus atributos, já os modificadores são equivalente ao seu “Valor” subtraído por dez. Os modificadores são o valor que mais tem peso dentro de jogo, sendo eles os valores usados para os cálculos e utilização do dito atributo.

Além das características informadas nestes tópicos, todos os atributos são relacionados a alguma perícia, podem ser utilizados por talentos e habilidades e podem ser pré-requisitos para a utilização de equipamentos e diálogos únicos com personagens não jogadores, que serão referidos daqui para frente como NPCs (*Non Player Characters*).

Talentos e habilidades também podem adicionar funções a um atributo.

6.3.1.1 Força

Dentro de jogo a força determina o dano físico causado pelo personagem, desarmado ou armado com armas que utilizem deste atributo, e é o atributo chave para testes de força. Também determina quanta carga o personagem consegue carregar em seu inventário.

6.3.1.2 Destreza

É a base para os cálculos de dano com armas que utilizem este atributo, também é o atributo chave para testes de destreza.

6.3.1.3 Agilidade

Determina o deslocamento total do personagem, sua velocidade de ataque, sua defesa física e é o atributo chave para testes de agilidade.

6.3.1.4 Constituição

É o atributo que define o valor de pontos de vida e energia do personagem, assim como é o atributo usado nos testes de constituição.

6.3.1.5 Inteligência

Um atributo usado para o cálculo de pontos de mana, dependendo do tipo de conjuração do personagem, além da quantidade de magias iniciais e o dano de magias relacionadas a esse atributo, e também influências a quantidade de perícias treinadas. Além de ser o atributo chave para testes de inteligência.

6.3.1.6 Sabedoria

É o atributo utilizado para o cálculo de pontos de mana e dano mágico, de acordo com o perfil de conjuração do personagem, quantidade de magias iniciais e dano de magias referentes à sabedoria, e é o atributo base para testes de sabedoria.

6.3.1.7 Carisma

É o atributo que estabelece a base para o cálculo de pontos de mana, de acordo com o perfil de conjuração do personagem, quantidade de magias iniciais e o dano de magias relacionadas a esse atributo, também é o principal atributo usado para determinar a efetividade de interações com NPCs e quão prestativos os mesmos são. Além de ser o atributo chave para testes de carisma.

6.3.1.8 Criatividade

Principal atributo responsável pela criatividade do personagem quando utilizar perícias de criação de itens e requisito para a pesquisa de atributos de itens, também concede um pequeno bônus para todas as perícias do personagem e aumenta a quantidade de experiência ganha dentro de jogo.

Para definir os atributos de seu personagem o jogador deve alocar sua pontuação inicial, que pode ser afetada pela raça de seu personagem, sendo ela: oito, dez, doze, quatorze, dezesseis e dezoito. Após alocar suas pontuações, estas poderão ser afetadas pela sua escolha de raça.

Durante o jogo, toda vez que seu personagem aumentar de nível, o jogador recebe um ponto para distribuir entre seus atributos.

6.3.2 Raça

A raça do personagem define, além da aparência e base de relacionamento com outras raças, bônus ao personagem e, de acordo com a raça, até mesmo penalidades. Neste tópico serão listadas as raças disponíveis em jogo para o jogador, sendo apresentadas com: suas características comuns, relações com outras raças, bônus que rendem ao personagem e penalidades (se houverem).

6.3.2.1 Humanos

Humanos são quase como a definição de variedade e adaptabilidade, existem com as mais diversas personalidades e tendências, diferenciando-se até mesmo entre a própria raça, de forma física e psicológica. Fisicamente falando, geralmente, um humano adulto tem no mínimo um metro e meio e, nos casos mais extremos, podem chegar até ter dois metros e meio. Mas como é característica da raça, toda regra tem suas exceções. Já em ideais e personalidades, tendem a ser variados e únicos, distinguindo-se em quase tudo que é possível, mesmo que em pequenos detalhes.

Um personagem de jogador da raça humana recebe dois talentos adicionais, dez pontos para distribuir entre seus atributos, sendo que é permitido adicionar no máximo cinco pontos a um único atributo, além disso, personagens humanos recebem mais duas perícias treinadas.

6.3.2.2 Elfos

São uma raça antiga, lendas dizem que foram uma das primeiras raças de Nova Ventis, que tem vidas muitas vezes mais longas que as humanas, possuem um vínculo muito forte com a magia. São sagazes, altivos e muito orgulhosos, normalmente beirando a arrogância. Fisicamente costumam ser mais altos que humanos, com feições mais esguias, de orelhas longas e pontuadas e cabelos lisos, e normalmente, longos.

Devido a sua natureza, tem dificuldades em lidar com outras raças, com exceção da raça humana.

Um personagem de raça élfica recebe um adicional de cinco pontos em destreza, três em inteligência e agilidade, e possuem uma penalidade de três pontos em constituição, possuem um bônus de mais quatro pontos em testes de resistência contra magias debilitantes, mais quatro pontos nas perícias de identificar magia e percepção. Além disso, sabem usar espadas longas e arcos (curtos, longos e compostos), e recebem foco em uma dessas armas.

6.3.2.3 Anões

Um povo recluso, resiliente, que sente prazer nos ofícios de criação e mineração, valorizando muito a força e habilidade com a forja. São um povo teimoso, às vezes até mesmo rabugento, desconfiado, com opiniões e ideais duros como rocha, quase imutáveis durante sua vida, mas como o metal que tanto amam, podem se moldar. Já suas características físicas, são um povo pequeno em estatura, mas muito robustos e pesados, com exceção das mulheres anãs que tendem a ser mais leves, desde jovens os anões geralmente cultivam barbas longas e grossas.

Levando em conta sua personalidade, anões são difíceis de lidar, tendo poucas raças como amigas, e devido a suas diferenças possuem uma notória inimizade com os elfos. Mas, como os humanos demonstraram, quando se conquista a aliança e confiança dos anões, ela é mais duradoura que o melhor dos metais.

Como anões personagens recebem bonificações de cinco pontos em constituição, três pontos em sabedoria e uma penalidade de três pontos em destreza e agilidade, também recebem um ofício a escolha entre Metalurgia, Joalheria e Mineração, esse ofício receberá um bônus de quatro pontos adicionais. Devido a sua composição física, anões recebem bônus de quatro pontos na defesa física contra oponentes de tamanho grande.

6.3.3 Árvores de habilidades

Neste sistema, os personagens podem se especializar em uma de três árvores de habilidades, essa especialização rende pontos adicionais nos níveis iniciais para o jogador adquirir novas habilidades ou melhorar, quando possível, as que já possui.

No primeiro nível os personagens possuem três pontos para utilizar entre todas as árvores e dois pontos adicionais para a árvore na qual é especializado.

Todas as habilidades podem ser adquiridas ou melhoradas utilizando um destes “pontos de habilidade”, e cumprindo seus pré-requisitos para fazê-lo, que podem envolver outras habilidades, o nível de personagem, certo valor em um ou mais atributos e até mesmo que certos objetivos sejam cumpridos.

As habilidades podem ser divididas em duas categorias gerais, ativas quando tem seus efeitos ativados apenas quando o jogador as utiliza ou passivas que tem seus efeitos sempre ativos e o jogador não precisa se preocupar em ativá-las.

As habilidades são influenciadas principalmente pelos atributos, mas em casos específicos podem ser influenciadas por perícias, e também podem influenciar atributos e perícias.

Cada árvore possui habilidades únicas, e são separadas por suas funções:

- **Combate:** é a árvore onde estão concentradas as habilidades focadas em combate, utilização de armas e armaduras, golpes especiais, posturas de combate e habilidades físicas de combate.
- **Magias:** nesta árvore se encontram todas as habilidades que lidam com magia, um personagem só pode utilizar habilidades mágicas se investir pontos nesta árvore.
- **Especialista:** por outro lado nesta se concentram habilidades que lidam com as perícias do personagem, funções fora de combate, vantagem estratégica e bônus passivos não relacionados a combate, normalmente.

O personagem recebe mais pontos de habilidade toda vez que adquire um novo nível, também podem ser obtidos ao completar missões, *quests* únicas, desafios ou atividades únicas.

6.3.4 Perícias

São habilidades que não envolvem combate direto, sendo utilizadas pelos jogadores para empenhar funções que independem das suas habilidades ou talentos, no entanto, utilizam como base algum atributo do personagem. As perícias disponíveis serão listadas abaixo junto de uma descrição, sua função e atributo chave:

- **Atletismo:** perícia utilizada para determinar as habilidades do personagem de desempenhar proezas atléticas, como escalar, correr, nadar e saltar. Em jogo atletismo pode ser utilizado para passar por obstáculos de terreno e em ações de corrida em combate. Tem como atributo chave a força.
- **Acrobacia:** tem como finalidade medir as habilidades do personagem em realizar façanhas que requerem um alto grau de controle corporal, agilidade, leveza e graciosidade, como andar numa corda bamba, realizar acrobacias, movimentar-se não previsivelmente. Dentro de jogo pode ser usada para superar obstáculos de terreno. O atributo chave desta perícia é a destreza.
- **Conhecimento:** mostra o quanto o personagem estudou algum assunto, normalmente sendo ligado a algum tipo de estudo acadêmico ou científico, esta perícia pode ser escolhida mais de uma vez devido às suas variações de conhecimentos disponíveis, sendo elas: arcanismo, religião, estratégia, natureza e história. Sendo assim, um personagem pode ter um número cumulativo dessa perícia igual a quantidade máxima de conhecimentos disponíveis. Suas funções em jogo são para, fora de combate, a resolução de mistérios, identificação de itens, missões e história, já dentro de combate esta perícia pode ser usada para identificar as estatísticas e características de um oponente,

baseado no conhecimento usado. A inteligência do personagem é o atributo chave dessa perícia.

- **Cura:** esta é utilizada para demonstrar o conhecimento do personagem no cuidado de ferimentos, doenças e venenos. Em jogo, cura pode ser usada para retirar efeitos negativos ou reduzir sua duração, curar o personagem dentro de combate e fora dele, para desempenhar essas funções são necessários certos itens. Esta perícia utiliza sabedoria como seu atributo chave.
- **Furtividade:** demonstra a habilidade do personagem de se mover silenciosamente, se esconder e esconder sua presença. Pode ser usado em jogo para passar despercebido por inimigos, ficar fora de seu foco de combate e atacar inimigos sem que eles o percebam. Utiliza o atributo destreza como chave.
- **Identificar magia:** é a habilidade do personagem de identificar fontes de energia mágica, natural ou não, seus efeitos e intensidade. Pode ser usada dentro de jogo para identificar propriedades mágicas de itens e de elementos do cenário. Emprega como atributo chave a inteligência.
- **Iniciativa:** representa a velocidade de resposta do personagem ao perigo. Sendo uma perícia crucial em combate, determinando a ordem dos turnos dos envolvidos de acordo com o maior valor de iniciativa entre eles. O atributo usado por esta perícia é a agilidade.
- **Ofício:** é a perícia responsável pela criação de itens, sua avaliação e, se possível, seu reparo e manutenção. Assim como conhecimento, a perícia ofício pode ser escolhida mais de uma vez para cada ofício disponível, sendo eles: metalurgia, joalheria, carpintaria, curtição e mineração. Dentro de jogo o personagem pode criar, estudar características, reparar e avaliar itens que estejam dentro do escopo de seu ofício. Inteligência é o atributo chave desta perícia.

6.3.4.1 Valores de perícia

As perícias tem seu valor total estipulado pela soma de seu atributo chave, com suas graduações e pontos recebidos de outras fontes, como talentos ou habilidades das árvores. As graduações das perícias são determinadas, inicialmente por quais são as perícias treinadas do personagem, e depois podem ser aumentadas através de missões e prática.

A quantidade de perícias treinadas iniciais é baseada na árvore de habilidades em que o personagem é especializado somado ao modificador de inteligência no momento da criação do personagem, e pode ser influenciada por outras fontes, como as perícias adicionais que um personagem humano recebe. Sendo assim, temos:

- A árvore de habilidades de combate rende um total de perícias treinadas igual ao modificador de inteligência do personagem mais um;
- A árvore de habilidades de magia oferece ao personagem a perícia Identificar Magia, e seu modificador de inteligência mais um em perícias treinadas a sua escolha;
- Já a árvore de especialista concede um total de quatro mais seu valor de inteligência em perícias treinadas;

Perícias treinadas tem um valor total igual a o nível do personagem e ao modificador da habilidade chave da perícia. Perícias não treinadas tem seu valor igual à metade do nível do jogador mais o modificador da habilidade chave da perícia.

O jogador também poderá adquirir pontos em perícias através de atividades únicas, missões e *quests* que completar.

6.3.4.2 Perícias em prática

Quando o personagem utilizar uma perícia, ou for lhe requerido utiliza-la, um nível de dificuldade numérico é estabelecido e este deve ser superado pelo valor da perícia usada mais um valor aleatório de um a quinze, se o valor da perícia for maior que o nível de dificuldade o valor aleatório é desconsiderado e

o personagem é bem sucedido no uso da perícia. Um exemplo: o jogador deseja que seu personagem pule de uma ponte a uma plataforma de pedra distante, o jogo lhe dará esta opção que será confirmada ao pressionar o botão de ação quando este for indicado, ao fazê-lo o jogo calcula uma dificuldade de valor quinze em um teste da perícia atletismo, o personagem possui a perícia atletismo treinada, seu nível é um e possui cinco pontos de modificador em força. Assim o personagem tem um valor total de seis mais o valor aleatório para superar a dificuldade estabelecida, se bem sucedido consegue realizar o salto se não, não conseguirá realizar o salto.

Já no caso de criações, reparo e pesquisa de itens, os níveis de dificuldade são calculados com base na complexidade e valor do item, sendo necessários os materiais para realizar tais ações. Em todos os casos ao passar do nível de dificuldade estabelecido para ação desejada, o personagem consegue realizá-la, no caso de falha o personagem será proibido de tentar a mesma ação com o mesmo item por um tempo, baseado na dificuldade do teste, mas caso o personagem falhe por cinco ou mais pontos a baixo do nível de dificuldade estipulado, os materiais são perdidos e ainda haverá a penalidade de tempo de espera.

6.3.5 Talentos

São como habilidades, só que de níveis mais básicos e gerais, que oferecem novas capacidades a um personagem, ou melhoram as que já possui. Podendo influenciar suas ações, perícias, habilidades e magias. Por exemplo, para que um personagem saiba utilizar uma armadura do modelo pesada, ele deve possuir o talento Armadura Pesadas, e caso o jogador deseje que seu personagem se torne ainda mais proficiente com esse tipo de armadura pode optar por adquirir o talento Foco em Armadura Pesada.

Talentos tem a necessidade de que o personagem cumpra os pré-requisitos do mesmo para que possa adquiri-lo.

A especialização do personagem em uma árvore de habilidades oferece pacotes de talentos iniciais, que o jogador poderá escolher.

Os talentos podem ser classificados em quatro grupos, sendo eles:

- Talentos de Combate: voltados para habilidades usadas dentro de combate;
- Talentos de Perícias: usados para dar mais usos as perícias existentes ou melhora-las;
- Talentos de Magia: fornecem vantagens, maior eficácia e melhorias as magias e recursos mágicos do personagem;
- Talentos Heroicos: são talentos únicos que um personagem só pode obter quando o jogador completar certas missões ou atividades únicas desempenhadas em sua vida real.

Além disso, o personagem recebe um talento adicional a cada quatro níveis, a partir do primeiro, que pode ser utilizado para adquirir novos talentos que tenham seus requisitos cumpridos.

6.4 MECÂNICAS IN-GAME

Aqui serão abordadas as mecânicas dentro de jogo, a par da criação de personagem, sendo divididas em: mundos e exploração, estatísticas de personagem, evoluções do personagem, combate, monstros, *Non Player Characters*, aventurando-se e regras.

6.4.1 Mundos e exploração

Os “mundos” são os mapas que o jogador terá acesso, no caso um dentro de jogo, separado em um gigantesco continente, sendo ele Nova Ventis, onde serão disponibilizadas as *quests*, história, *dungeons*, aventuras e exploração dentro de jogo. Fora de jogo, utilizando da geolocalização, Vuforia e *Mapbox*, os mapas disponíveis de nosso mundo real serão usados para abrigar

desafios, *dungeons*, aventuras, encontros aleatórios, localização de outros jogadores e eventos, em localizações reais em tempo real.

Para realizar a exploração dos mapas disponíveis dentro de jogo o jogador terá de aceitar *quests*, encontrar ganchos de aventuras ou localizar *dungeons* através de itens, da história do jogo ou de *quests*. Além disso, localizações comuns já estarão disponíveis para exploração. O acesso dessas áreas se dá através do mapa do continente onde estas áreas comuns estarão disponíveis, e sempre que o jogador descobrir novas, elas serão mostradas no mapa e poderão ser exploradas. As localizações dentro de jogo só podem ser acessadas uma vez para o mesmo tipo de exploração em um certo intervalo de tempo, determinado pelo nível da área.

Já na exploração do mapa baseado em nosso mundo, o jogador tem acesso a todos os locais que pode ir, podendo encontrar monstros em quase qualquer local, exceto em certos tipos de áreas. Áreas tem certos tipos determinados pelos locais onde se encontram, os tipos de áreas disponíveis e como funcionam são:

- **Áreas de treinamento:** são locais públicos, como praças e parques, normalmente que possuam reputação ou sejam apontadas pelos dados do *Mapbox* como áreas de práticas de exercícios. Nestes locais o jogador recebe bônus de experiência em seus atributos físicos para atividades realizadas durante sua estadia nesta área. Jogadores em grupos recebem bonificações ainda maiores;
- **Áreas de comércio:** localidades publicamente conhecidas como comércio, como praças no distrito comercial da cidade ou shoppings, nestas áreas os jogadores podem deixar lojas abertas, como pequenas barracas onde seus personagens permanecem para vender itens que estiverem em seu inventário de loja. Nestes locais os jogadores também recebem bônus em experiência para atividades que envolvam carisma e recebem experiência neste mesmo atributo quando interagem com jogadores ou fazem novos amigos dentro de jogo;

- **Áreas de estudo:** são localidades de estudo, como bibliotecas e livrarias, onde os jogadores possam estudar, praticar seu conhecimento, praticar seus ofícios reais e/ou debater. Devido a sua natureza áreas de estudo podem ser abertas por grupos com três jogadores ou mais. Nestas áreas os jogadores recebem bonificações de experiência em atividades que envolvam inteligência, sabedoria e criatividade;
- **Áreas de evento:** localizações onde ocorrem eventos promocionais do jogo, ou de empresas que comprem licenças para serem marcadas como áreas elegíveis para eventos, onde ocorrem atividades especiais como aparecimento de monstros lendários, caravanas de *NPCs* que vendem itens raros, exploração de cavernas do tesouro entre outros;
- **Dungeons:** são localidades públicas ou quando privadas, que os jogadores possam acessar, como prédios públicos ou áreas declaradas como *dungeons*, nestas áreas os jogadores deverão formar grupos de no mínimo quatro jogadores para acessar seus conteúdos e desafios dentro de jogo.

6.4.2 Estatísticas de personagem

Aqui serão esclarecidos os termos utilizados durante este projeto como características do personagem, como defesa, pontos de vida, pontos de mana e entre outros.

Estes são atributos secundários do personagem, que são influenciados por atributos, habilidades, talentos, itens, magias ou até mesmo por efeitos de cenário. Sendo eles:

- **Pontos de vida:** são usados para medir quanto dano o personagem pode receber. Quando os pontos de vida de um personagem chegam a zero ele cai inconsciente, caso isto ocorra dentro de combate o personagem é considerado derrotado por seus inimigos e recebe penalidades em sua quantidade total de experiência. Para recuperar pontos de vida o personagem deve descansar fora de combate, receber tratamentos de outros personagens (*NPCs* ou não), consumindo itens

que tenham tal efeito e/ou usando habilidades, talentos ou magias. Já dentro de combate o personagem pode recuperar pontos de vida apenas usando de itens, talentos, magias ou habilidades que tenham tal efeito;

- **Pontos de mana:** mede o total de magias o personagem pode utilizar, toda magia utiliza uma quantidade de pontos de magia baseado no seu nível, por exemplo, uma magia de primeiro nível consome um ponto de mana. Para recuperar pontos de mana o personagem pode descansar, utilizar itens, talentos e/ou habilidades que possam desempenhar essa função;
- **Pontos de vigor:** são utilizados por certas habilidades de combate, que os consomem para poderem ser utilizadas, diferentes de pontos de mana eles podem ser recuperados automaticamente quando um combate é finalizado, no entanto, durante o combate o personagem pode utilizar de uma ação completa para recuperar seu modificador de constituição em pontos de vigor;
- **Defesa física:** é a defesa utilizada contra ataques físicos, sendo determinada pelo equipamento utilizado pelo personagem e seu modificador de agilidade, também pode ser influenciada por outros itens, habilidades, talentos ou magias. Certas habilidades e talentos podem possibilitar que outros atributos influenciem a defesa física do personagem;
- **Defesa mágica:** é a defesa utilizada contra ataques mágicos, estabelecida pelo equipamento do personagem e seu modificador de conjuração, também pode ser manipulada por itens, habilidades, talentos ou magias. Certas habilidades e talentos podem permitir que outros atributos do personagem manipulem sua defesa mágica;
- **Teste de resistência:** são utilizados com base em um atributo, para resistir efeitos nocivos ou danoso, normalmente provindos de armadilhas, itens, habilidades ou magias;
- **Testes de atributo:** são praticados quando um personagem realiza uma ação ou sofre a ação de outro, que lhe impõe um teste baseado apenas

em seu atributo. Por exemplo: quando um personagem tenta empurrar algo muito pesado lhe é requerido um sucesso em um teste de força para empurrar o objeto;

- **Carga:** designa o total de peso que o personagem pode suportar, englobando itens que ele esteja utilizando ou que estão dentro de seu inventário. A carga possui três níveis: leve, média e pesada. Até o nível de carga leve o personagem não sofre nenhum tipo de penalização, ao passar seu valor máximo de carga leve ele entra em carga média, onde seu deslocamento é reduzido em três metros e sua iniciativa é reduzida em três pontos, e quando alcança o nível de carga pesada ele já não pode carregar mais nenhum item e perde metade do seu deslocamento e reduz sua iniciativa em seis pontos;
- **Deslocamento:** é a medida de quantos metros um personagem pode se deslocar em uma ação de movimento, ou três segundos, sendo padronizada em nove metros, mas personagens cujo modificador de agilidade ultrapasse nove, passa a utilizar o valor de seu modificador como deslocamento;
- **Atributo de conjuração:** é o atributo que o personagem utiliza para conjurar magias, é determinado através da habilidade inicial da árvore de habilidades mágicas.

6.4.2.1 Evolução do personagem

Um personagem tem sua evolução medida através de seu nível e de seus atributos, estes são aumentados de acordo com o acúmulo de experiência que o personagem possui. Existindo dois tipos dela: a experiência de personagem e a experiência de atributos.

Sempre que um jogador pratica uma atividade na vida real que lhe bonifique com experiência a um atributo, esta se acumula e quando alcançam certas quantidades, o atributo é aumentado em uma determinada quantidade de pontos.

Dentro de jogo, o personagem acumula experiência para si, com o intuito de aumentar seu nível, todo nível oferece algum tipo de bonificação ao personagem.

Em ambos os casos sempre que o personagem alcança a quantidade de experiência para o próximo nível, a quantidade para o próximo é sempre maior.

6.4.3 Combate

Quando um personagem de jogador chega a uma determinada distância pode iniciar um combate com um monstro, atacando ou sendo atacado. Este primeiro ataque é transferido para um turno extra dentro de combate, que pode ou não acertar seu alvo. Como indicado, os combates são divididos em turnos, com um ataque extra sem eu inicio se o jogador ou monstro aproveitarem a oportunidade, e após esse turno extra ter ocorrido ou não, é determinada a ordem de iniciativa, estipulada pelo valor de iniciativa de cada individuo envolvido no combate.

Um turno é composto por uma ação padrão, uma ação de movimento, uma ação simples e ações rápidas. Uma ação padrão é usada para atacar, equipar itens do inventário do personagem, utilizar uma perícia que necessite desta ação ou pode ser transformada em uma ação de movimento extra. Já uma ação de movimento é utilizada pelo personagem para sacar uma arma já equipada ou movimentar-se utilizando seu deslocamento. A ação simples normalmente é utilizada por habilidades ou talentos que necessitem ser ativados, e as ações rápidas são operações rápidas que não possuem limite e podem ocorrer indeterminadas vezes em um combate, são provenientes de habilidades, certas perícias e talentos.

Outra ação possível é a ação de turno completo, que consome todo o turno para realizar determinada uma determinada função, proveniente de algum tipo de magia, habilidade ou talento.

Durações de efeitos são medidas em números de rodadas dentro de combate, uma rodada é um círculo completo entre os participantes do embate.

6.4.3.1 Ataques e dano

Ataques são realizados utilizando de magias, golpes armados ou desarmados. Quando um personagem ataca outro, este deve realizar sua jogada de ataque, utilizando o atributo chave para este baseado na forma deste ataque somado a um valor aleatório entre um e vinte, se a soma de valor superar o tipo de defesa respectivo ao tipo de ataque sendo realizado o atacante acerta o ataque e causa dano, senão o erra e consome sua ação da mesma forma.

O dano é determinado somando o atributo chave da forma de ataque mais as estatísticas dessa forma de ataque, que normalmente variam de um dano mínimo a um dano máximo. Por exemplo: o personagem do jogador acerta o ataque em um monstro, utilizando uma espada longa que lhe dá um adicional ao dano de um a dez e possui cinco pontos no modificador de força que é utilizado como atributo chave por sua arma, assim ele causará um dano que pode variar de seis a quinze.

Contudo, ataques possuem um valor de chance de acerto crítico, que seria o número de valores, a partir do número vinte, causam o multiplicador de dano crítico de sua forma de ataque. Usando do mesmo exemplo apresentado acima: a espada longa do jogador possui uma chance de acerto crítico de dois e um multiplicador de dois no dano crítico, ou seja dezoito e vinte no valor aleatório, caso o valor aleatório seja um destes números ele causará duas vezes o dano de seu ataque, provindo do multiplicador de dano crítico da arma.

Uma regra extra é que todo valor um no valor aleatório somado a ataques é sempre um erro e um valor vinte é sempre um acerto crítico.

6.4.3.2 Magias

Magias possuem quatro características determinadas: tempo de conjuração, tempo de duração, efeito e nível. Tempo de conjuração é o tipo de ação que a magia consome para ser utilizada, tempo de duração estipula por quanto tempo o efeito da magia permanece, podendo ser instantânea ou durar até mesmo minutos, o efeito é uma descrição que explica quais os efeitos da magia, se

causa dano, debilita seus inimigos ou concede bônus ou vantagens ao personagem, e o nível designa o nível da magia e seu custo em pontos de mana.

Para que um jogador possa aprender uma magia ele deve superar seu nível ou iguala-lo com o modificador de seu atributo de conjuração.

Magias podem ter níveis zero, ou seja, não consomem nenhum ponto, estas são chamadas de truques e são consideravelmente mais fracas que magias que consomem pontos de mana.

6.4.4 Monstros

São as criaturas criadas e controladas pelo sistema do jogo, dentro do “mundo real” eles aparecem em geolocalizações aleatórias, mas tendem a aglomerar-se ao redor de *dungeons*, áreas de eventos e de treinamento. Já no mundo dentro de jogo, eles estão distribuídos pelas áreas exploráveis pelos jogadores, fora de áreas seguras, como vilas, castelos ou fortalezas.

Dentro de *dungeons* e áreas especiais os monstros tendem a ser mais fortes e um “chefão”, um monstro único e mais forte, espreita o final destas localizações.

Derrotar monstros é uma das formas que o personagem do jogador pode receber experiência e encontrar itens.

6.4.5 Non Player Characters

Personagens criados e controlados pelo sistema do jogo são chamados de *Non Player Characters* (NPCs), estes estão dentro de jogo para apresentar a história dele ao jogador, dar-lhe missões, *quests*, aventuras, ensinar habilidades ou talentos e comprar ou vender itens. Além de poderem aparecer em eventos aleatórios pelo mapa de nosso mundo ou em áreas de eventos ou comércio.

6.4.6 Aventurando-se

As *quests* e aventuras são missões dentro de jogo que contam histórias e recompensam os jogadores quando as completam.

Quests são missões comuns, como coletar matérias ou derrotar monstros, podem ser adquiridas através de diálogos com NPCs, itens ou até mesmo de jogadores. Quando jogadores criam *quests* elas ficam disponíveis em áreas de comércio e treinamento, para que outros jogadores possam visualiza-las e se quiserem aceita-las.

Aventuras são missões longas, mais complexas, que costumam envolver NPCs e histórias criadas para elas. Podem ser adquiridas em áreas de eventos, encontrando itens que servem de gancho para a aventura e através do diálogo com NPCs.

6.4.7 Itens

Os itens existem em categorias, cada uma delas possuindo uma funcionalidade diferente, são elas:

- **Equipamentos:** são itens que o personagem pode utilizar como equipamento, como armas ou armaduras;
- **Consumíveis:** designam aqueles itens que para desempenharem seus efeitos devem ser consumidos pelos jogadores, normalmente possuem um valor total de usos antes de se esgotarem;
- **Materiais:** englobam os itens que são usados de matérias primas para os ofícios de criação;
- **Itens de missão:** indicam os itens que são criados pelo jogo como objetivos de missões, que serão utilizados apenas para isso.

6.4.8 Regras

As regras implícitas são galgadas nas mecânicas de jogo, já as regras apresentadas formalmente ao jogador são:

1. Não desrespeitara ou será ofensivo a outro jogador;

2. Todo e qualquer tipo de preconceito será tratado como ofensa grave, levando a denúncia do jogador que os praticar e expulsão imediata após a análise deste ato;
3. Jogadores que utilizam de programas de terceiros para manipular e trapacear serão expulsos do jogo;
4. *Nova Ventis Adventurers* baseia-se totalmente no *honor play*, ou seja, da honra dos que o jogam, devido a seu objetivo. Apesar de jogadores que sejam denunciados serão expulsos, partimos da suposição de que os jogadores jogaram em busca de se melhorar e evoluírem com seus personagens.

7 CONSIDERAÇÕES FINAIS

O desenvolvimento desta pesquisa possibilitou a análise e desenvolvimento de novas perspectivas de utilização da gamificação em jogos por meio do uso de realidade aumentada e mecânicas que buscam aumentar o engajamento do jogador em seu autodesenvolvimento e relacionamento social, apresentando perspectivas de desenvolvimento, conceitos e técnicas para aplicação das mesmas. A pesquisa fez uso de estudos e mecânicas previamente existentes, e procurou estabelecer novas mecânicas, para fundamentar tais conceitos.

As novas técnicas e mecânicas para a aplicação da gamificação da vida utilizam de RA para aprofundar ainda mais sua imersão e aplicação na vida do usuário.

Além disso, a pesquisa trouxe um arcabouço teórico robusto para ser utilizado como base no desenvolvimento e aplicação dos conceitos estudados, bem como uma introdução às ferramentas Unity e Vuforia e como elas podem ser aplicadas em um jogo de realidade aumentada.

8 REFERÊNCIAS

Android Developers, **Room Persistence Libray**. Disponível em: <<https://developer.android.com/topic/libraries/architecture/room.html>>. Acesso em: Março de 2018.

Android Source, The Android Source Code. Disponível em:

<<https://source.android.com/setup/>>. Acesso em: Março de 2018.

AMIR, Bilal; RALPH, Paul. **Proposing a theory of gamification effectiveness**. In: Companion Proceedings of the 36th International Conference on Software Engineering. ACM, 2014. p. 626-627.

BERGAMASCHI, M.; DE MORAIS, Thamires Martins Augusto. **Estudo sobre a utilização de VUFORIA e Unity 3D com RA para dispositivos móveis**. 2014.

CASSARO, Marcelo, Tormenta RPG: edição revisada / Guilherme Dei Svaldi e Leonel Caldela; ilustrações por Erica Awano [et al.]. -- -- Porto Alegre: Jambô, 2013.

DE OLIVEIRA, Thiago HB et al. Introdução a Modelagem 3D com Blender 3D. **Revista da Escola Regional de Informática**, v. 2, n. 2, p. 202-209, 2013.

KIRNER, Claudio; SISCOOTTO, Robson. Realidade virtual e aumentada: conceitos, projeto e aplicações. In: **Livro do IX Symposium on Virtual and Augmented Reality, Petrópolis (RJ), Porto Alegre: SBC**. 2007.

Library Vuforia, **Smart Terrain**. Disponível em:

<<https://library.vuforia.com/articles/Training/Getting-Started-with-SmartTerrain>>. Acesso em: 17 de Março de 2018.

MEGALI, Tin. **Realidade Aumentada Estilo Pokémon GO Com Vuforia**. Disponível em:

<<https://code.tutsplus.com/tutorials/introducing-augmented-reality-with-vuforia--cms-27160>>. Acesso em: 18 de Março 2018.

NAVARRO, Gabrielle. **Gamificação: a transformação do conceito do termo jogo no contexto da pós-modernidade**. Biblioteca Latino-Americana de Cultura e Comunicação, v. 1, n. 1, p. 1-26, 2013.

OWENS, Mike; ALLEN, Grant. **SQLite**. Apress LP, 2010.

PALLADINO, Tommy. **Mapbox lança plataforma AR baseada em localização com suporte multiusuário**. Disponível em:

<<https://next.reality.news/news/mapbox-launches-location-based-ar-platform-with-multi-user-support-0183137/>>. Acesso em: 18 de Março de 2018.

ROVADOSKY, Douglas Samuel et al. Uma aplicação de realidade aumentada para dispositivo móvel com sistema operacional Android. **Revista Brasileira de Computação Aplicada**, v. 4, n. 1, p. 25-37, 2012.

SILVA, Bruna Sena da; LEMOS, Lays Ribeiro; QUEIROZ, Lygia Cristina Soares de Brito de; NODA, Melissa Sheila Monteiro. **O Estado da Arte Brasileira sobre Realidade Aumentada**. Disponível em: <<https://pt.scribd.com/document/101051935/O-Estado-Da-Arte-Brasileira-Sobre-Realidade-Aumentada>>. Acesso em: Outubro de 2017.

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

Porto Alegre: Jambô, 2013. Totalcrossblog. **Geolocalização: conceitos e aplicações**. Disponível em:

<<http://www.totalcross.com/blog/geolocalizacao-conceitos-e-aplicacoes/>>. Acesso em: 17 de Março de 2018.

The Knights of Unity, **Unity AR Tutorial: Augmented Reality Game Development with Vuforia**. Disponível em: <<https://blog.theknightsofunity.com/unity-vuforia-guide/>>. Acesso em: 25 de Setembro de 2018.

Unity3D, **Plugins**. Disponível em: <<https://docs.unity3d.com/Manual/Plugins.html>>. Acesso em: 14 de Março 2018.

Unity3D, **Unity3D**. Disponível em:

<<https://unity3d.com/pt/unity/features> >. Acesso em: 16 de Março 2018.

VERGARIA, Mauricio. **Vuforia in Unity: Build cross-platform AR apps**. Disponível em: <<https://blogs.unity3d.com/pt/2018/01/15/vuforia-in-unity-build-cross-platform-ar-apps>>. Acesso em: Março 2018.

VOGEL, Lars. Android development tutorial. **Based on Android**, 2013.

Vuforia, **Features**. Disponível em: <<https://www.vuforia.com/features.html>>. Acesso em: 16 de Março de 2018.