

CENTRO UNIVERSITÁRIO ATENAS

PEDRO HENRIQUE RIBEIRO SILVA

INTELIGÊNCIA ARTIFICIAL APLICADA A JOGOS DIGITAIS

PARACATU

2019

PEDRO HENRIQUE RIBEIRO SILVA

INTELIGÊNCIA ARTIFICIAL APLICADA A JOGOS DIGITAIS

Monografia apresentada ao Curso de Sistemas de Informação do Centro Universitário Atenas, como requisito parcial para obtenção do título de Bacharel em Sistemas de Informação.

Área de Concentração: Jogos Digitais

Orientadora: Prof^a. Msc. Raranna Alves da Costa

PARACATU

2019

PEDRO HENRIQUE RIBEIRO SILVA

INTELIGÊNCIA ARTIFICIAL APLICADA A JOGOS DIGITAIS

Monografia apresentada ao Curso de Sistemas de Informação do Centro Universitário Atenas, como requisito parcial para obtenção do título de Bacharel em Sistemas de Informação.

Área de Concentração: Jogos Digitais

Orientadora: Prof^a. Msc. Raranna Alves da Costa

Banca Examinadora:

Paracatu – MG, _____ de _____ de _____.

Prof^a. Msc. Raranna Alves da Costa
Centro Universitário Atenas

Prof. Msc. Felipe Neto Vasconcelos
Centro Universitário Atenas

Prof. Douglas Gabriel Pereira
Centro Universitário Atenas

RESUMO

O presente trabalho tem como objetivo demonstrar como a inteligência artificial aplicada a um jogo pode aumentar o engajamento dos jogadores. Para isso entendeu-se que ao aplicar uma inteligência artificial no jogo esta inteligência traz novos tipos de comportamento para um único personagem não jogável. A fim de validar tais afirmações foi necessário o estudo do assunto por meio de uma metodologia exploratória, ao fim do estudo foi constatado como uma inteligência artificial irá se comportar e quais impactos ela gerou dentro do jogo. Para melhor compreensão de como todo o processo foi feito, primeiramente tem-se uma leve explicação sobre outros temas que são essenciais para a finalização do trabalho, a fim de obter o conhecimento necessário para maior compreensão do leitor. Os jogos digitais são considerados softwares que promovem a interação do jogador para realizar escolhas assim podendo atingir os objetivos em meio a regras pré-definidas. A utilização de uma inteligência artificial em jogos digitais não é novidade, seu uso começou em jogos de xadrez, mas no mercado atual é difícil encontrar um jogo que não utilize alguma técnica de inteligência artificial. Para desenvolvimento de jogos existem vários motores de games disponíveis no mercado, um que se destaca dentre os demais é o unity que nos permite criar jogos em 2D e 3D, conseguindo exportar a aplicação para qualquer outra plataforma.

Palavras-chave: Jogos digitais. Inteligência Artificial. Unity.

ABSTRACT

This paper aims to demonstrate how artificial intelligence applied to a game can increase or engage players. To this end, understand that by applying artificial intelligence to the game, intelligence brings new types of behavior to a single unplayable character. An end of validation of such statements was necessary for the study of means of an exploratory methodology, and for the end of a study it was verified how an artificial intelligence to behave and what consequences it generated within the game. For a better understanding of how the whole process was done, first we have a light explanation about other topics that are essential for the finalization of the work, in order to gain the knowledge necessary for a better understanding of the reader. Digital games are considered software that promotes the interaction of the player to make choices so that they can achieve their goals through predefined rules. The use of artificial intelligence in digital games is not new, its use began in chess games, but in today's market it is difficult to find a game that does not use any artificial intelligence technique. For game development there are several game engines available in the market, one that stands out among the others is unity that allows us to create games in 2D and 3D, being able to export the application to any other platform.

Keywords: Digital game. Artificial intelligence. Unity.

LISTA DE FIGURAS

| | |
|--|----|
| FIGURA 1 - Interface do unity | 20 |
| FIGURA 2 - Scene view | 20 |
| FIGURA 3 - Game view | 21 |
| FIGURA 4 - Hierarchy | 22 |
| FIGURA 5 - Inspector | 22 |
| FIGURA 6 - Project | 23 |
| FIGURA 7 - Tela inicial | 31 |
| FIGURA 8 - Fase 1 | 32 |
| FIGURA 9 - Fase 2 | 33 |
| FIGURA 10 - Tela de fim de jogo | 34 |

SUMÁRIO

| | |
|--|-----------|
| 1 INTRODUÇÃO | 8 |
| 1.1 PROBLEMA DE PESQUISA | 8 |
| 1.2 HIPÓTESES | 8 |
| 1.3 OBJETIVOS | 9 |
| 1.3.1 OBJETIVO GERAL | 9 |
| 1.3.2 OBJETIVOS ESPECÍFICOS | 9 |
| 1.4 JUSTIFICATIVA | 9 |
| 1.5 METODOLOGIA | 9 |
| 1.6 ESTRUTURA DO TRABALHO | 10 |
| 2 A INFLUÊNCIA DA INTELIGÊNCIA ARTIFICIAL PARA JOGOS DIGITAIS | 11 |
| 2.1 OS JOGOS DIGITAIS | 11 |
| 2.2 INTELIGÊNCIA ARTIFICIAL (IA) | 12 |
| 2.3 A IA E OS JOGOS DIGITAIS | 13 |
| 2.4 TÉCNICAS DE GAME AI UTILIZADAS EM JOGOS DIGITAIS | 14 |
| 2.4.1 MÁQUINAS DE ESTADOS FINITOS | 14 |
| 2.4.2 ALGORITMOS DE PATHFINDING | 15 |
| 2.4.3 ALGORITMOS GENÉTICOS | 15 |
| 2.4.4 SISTEMAS BASEADO EM REGRAS | 15 |
| 3 BENEFÍCIOS DE SE USAR UMA INTELIGÊNCIA ARTIFICIAL NOS JOGOS | 17 |
| 3.1 PROBLEMAS DE INTELIGÊNCIA ARTIFICIAL NOS JOGOS | 17 |
| 4 MOTOR DE JOGOS | 19 |
| 4.1 UNITY | 19 |
| 4.2 ESTRUTURA DO UNITY | 19 |
| 4.2.2 INTERFACE | 19 |
| 4.2.3 SCENE | 20 |
| 4.2.4 GAME | 21 |
| 4.2.5 HIERARCHY | 21 |
| 4.2.6 INSPECTOR | 22 |
| 4.2.7 PROJECT | 22 |
| 4.2.8 GAME OBJECTS | 23 |
| 4.2.9 ASSETS | 24 |

| | |
|---|----|
| 4.2.10 SPRITES | 24 |
| 4.2.11 SCRITPS | 24 |
| 4.2.12 SISTEMA DE FÍSICA | 25 |
| 4.3 C# | 25 |
| 4.4 APRESENTAÇÃO DO JOGO FRUIT SHOTING | 25 |
| 4.5 USO DE ASSETS | 26 |
| 5 CONSIDERAÇÕES FINAIS | 27 |
| REFERÊNCIAS | 29 |
| APÊNDICE A - Tela inicial | 31 |
| APÊNDICE B - Tela de jogo | 32 |
| APÊNDICE C - Segunda tela de jogo | 33 |
| APÊNDICE D - Tela de fim de jogo | 34 |

1 INTRODUÇÃO

Os jogos digitais são softwares que tem como finalidade tanto o aprendizado como o entretenimento. Segundo Schuytema (2008), para a existência de um jogo deve-se ter um universo fictício e paralelo ao real, com regras e limitações definidas pelo sistema, onde o usuário permanece inerte de forma lucida.

Neste enredo, pode-se definir Inteligência artificial (IA), como a capacidade de programar o computador para desempenhar tarefas que o pensamento humano e animal é capaz de fazer naturalmente (MILLINGTON, 2006).

Para Brian Schwab (2009) o desenvolvimento de jogos com Inteligência Artificial é vinculado principalmente aos comportamentos que a aplicação tem ao contato humano, à interação entre a interface e o usuário, mesmo que os algoritmos que controlam as interfaces, sejam os responsáveis pela inteligência.

Para que um jogo possa ter algum tipo de inteligência artificial é necessário utilizar técnicas e algoritmos de IA assim dando uma serie de comportamentos para o agente inteligente. Estas técnicas e algoritmos, de acordo com Bourg (2004), são utilizados no processo de criação dos jogos digitais para que os agentes tenham inteligência e personalidade.

1.1 PROBLEMA DE PESQUISA

Como a inteligência artificial por meio dos jogos digitais pode aumentar o engajamento dos jogadores?

1.2 HIPÓTESES

a) Acredita-se que com a implementação da inteligência artificial o jogo torna-se mais divertido e menos previsível, os agentes dotados com a inteligência artificial podem estar simulando comportamento humano, assim fazendo pensar que o jogador está interagindo com uma comunidade de outros jogadores dentro de um jogo.

b) Sem a implementação de uma inteligência artificial voltada a jogos os agentes do jogo estão sujeitos a um padrão que devem seguir. Já com sua implementação os agentes possuem a possibilidade de fazer escolhas, assim gerando um novo impacto no jogo que pode ou não mudar seu resultado final.

1.3 OBJETIVOS

1.3.1 OBJETIVO GERAL

Demonstrar como a implementação de uma IA pode aumentar o engajamento dos jogadores em um jogo.

1.3.2 OBJETIVOS ESPECÍFICOS

- a) demonstrar como uma IA se comporta dentro de um jogo.
- b) demonstrar os benefícios do uso de uma IA dentro de um jogo.
- c) destacar a criação de um jogo usando a ferramenta Unity.

1.4 JUSTIFICATIVA

A inteligência artificial está presente em todos os jogos da atualidade, seu uso se faz importante para que os agentes de um jogo mostrem algum comportamento inteligente seja para ajudar ou gerar obstáculos para o jogador. O seu uso também traz benefícios quando utilizado em jogos que exijam uma grande quantidade de comportamentos elaborados pelo agente inteligente, assim utilizando de algoritmos de aprendizado, estes agentes conseguem aprender e elaborar ações sem a necessidade do programador estar passando cada instrução que o agente inteligente deve fazer, assim poupando tempo durante a etapa de desenvolvimento do jogo.

Os jogos digitais por serem produtos comercializados, tem um prazo curto para desenvolvimento, para que esse desenvolvimento seja feito de forma mais ágil e eficaz os desenvolvedores optam por utilizar motores de jogos, assim facilitando todo o desenvolvimento pois o programador estará mais centrado em apenas desenvolver as regras e comportamentos que serão utilizados no jogo.

1.5 METODOLOGIA

A metodologia tem como objetivo descrever o tipo de pesquisa realizada para o trabalho de conclusão de curso ou para artigos científicos. Segundo Rampazzo (2005) é importante que um trabalho científico, seja desenvolvido por meio de uma sequência de métodos. Para o desenvolvimento do trabalho será utilizada a metodologia exploratória, que visa proporcionar maior conhecimento através de pesquisas bibliográficas e web de sites como unity e microsoft. De acordo com

Marconi e Lakatos (2003) pesquisa exploratória é a análise e uma pesquisa experimental que tem a função e a elaboração de um problema e a busca pela sua solução. Esta metodologia oferece três propósitos: aumentar a familiaridade com o estudo, desenvolver as hipóteses do pesquisador com ideias e fatos, através de um ambiente para a elaboração de uma pesquisa futura.

1.6 ESTRUTURA DO TRABALHO

O trabalho de conclusão de curso (TCC) se divide em cinco capítulos, sendo que o primeiro capítulo apresenta a introdução, problema de pesquisa, hipótese, objetivos gerais, justificativa e a metodologia de estudo.

No segundo capítulo é apresentado a influência da inteligência artificial para os jogos, as técnicas de IA utilizar em jogos e os algoritmos de IA usados nos jogos.

O terceiro capítulo fala sobre os benefícios e problemas de se usar uma inteligência artificial em jogos digitais.

O quarto capítulo fala sobre a ferramenta unity usada para programar jogos digitais e uma breve apresentação do jogo desenvolvido pelo autor.

O quinto capítulo apresenta as considerações finais.

2 A INFLUÊNCIA DA INTELIGÊNCIA ARTIFICIAL PARA JOGOS DIGITAIS

Este capítulo explica sobre temas como jogos digitais, a inteligência artificial e como os dois interagem entre si, também é apresentado algumas técnicas de inteligência artificial utilizadas em jogos digitais presente no mercado.

2.1 OS JOGOS DIGITAIS

Os jogos digitais são um dos grandes atrativos da sociedade, seja ele usado para educação ou diversão. Para definir o que é um jogo Lucchese e Ribeiro (2009) nos fala que um jogo é uma sequência de atividades complexas, as quais inicialmente defendida por Huizinga em 1938, ao qual trazia a seguinte definição.

Um jogo pode ser definido como uma atividade livre, conscientemente tomada como “não-séria” e exterior à vida habitual, mas ao mesmo tempo capaz de absorver o jogador de maneira intensa e total. É uma atividade desligada de todo e qualquer interesse material, com a qual não se pode obter qualquer lucro, praticada dentro de limites espaciais e temporais próprios, segundo uma certa ordem e certas regras (HUIZINGA, 1938, p.10);

Essa definição ainda prevalece, incluindo os conceitos de jogos manuais, jogos físicos, e mais tarde a introdução aos jogos digitais. Os jogos digitais foram criados com intuito de ampliar e treinar a mente, com isso surgiram vários jogos, como jogos de raciocínio lógico e jogos educativos focados no ensino de disciplinas escolares como português e matemática básica (LUCCHESI, RIBEIRO, 2009).

De acordo com (JUUL, 2005), os jogos podem ser classificados em duas categorias de acordo com os desafios propostos, sendo jogos emergentes ou progressivos. Nos jogos emergentes, os desafios são regras simples com pequenos números que quando juntos formam uma grande variação de jogos, que necessitam de estratégias específicas, como os jogos de cartas e tabuleiro. Já os jogos progressivos contam com os objetivos em uma sequência de ações que os jogadores devem seguir.

Para (JUUL,2005) um jogo só pode ser caracterizado como tal, se satisfazer os seguintes requisitos:

- Ser um sistema formal baseado em regras,
- Com resultados variáveis e quantificáveis,

- Em que a cada resultado é possível associar valores distintos,
- Onde os jogadores despendem esforços para influenciar no resultado,
- Se sentem emotivamente ligados aos resultados,
- E as consequências de sua atividade são opcionais e negociáveis.

Para se definir o gênero que o jogo possui deve-se analisar seu objetivo principal, mesmo que com o passar dos anos os jogos tenham fragmentado esta ideia se dividindo em gênero e subgênero (ROGERS, 2010). Um jogo de ação é caracterizado pelo fato de o jogador ter respostas de reflexos e raciocínio rápido, esse por sua vez pode ser dividido em subgêneros como plataforma, envolver elementos de aventura e combates. Com regras e objetivos definidos um jogo pode ter uma finalidade adequada, seja educação ou entretenimento, com tudo a inteligência artificial é o que dará vida aos elementos do jogo, assim gerando desafios para o jogador (SCHUYTEMA, 2008).

2.2 INTELIGÊNCIA ARTIFICIAL (IA)

Quando se fala em IA logo se pensa em robôs que agem como humanos, para entender o que é inteligência artificial Russel (2002), apresenta diversas propostas para definir o que é inteligência artificial, dentre estas propostas observa-se como ideia fundamental a capacidade de aprendizagem e a manifestação de comportamento inteligente.

Para Russel e Norvig (2002), o campo da inteligência artificial está relacionado as dimensões centradas nos seres humanos e as racionalistas. A partir dessas dimensões Russel descreve quatro categorias para inteligência artificial sendo eles: sistemas que agem como seres humanos, sistemas que pensam como seres humanos, sistemas que pensam racionalmente e sistemas que agem racionalmente. A partir dessas categorias, se concretizaram testes onde se confirma se máquinas podem demonstrar comportamento e pensar como um humano.

Dentro das áreas de IA, como: Redes Neurais, sistemas especialistas, lógica incerta e outros, a técnica de busca ganha destaque por estar presente em aplicações distintas de softwares com propósitos diferentes. Segundo Dalmau (2004), o que pode se notar é a variedade de algoritmos de busca e a incidência desses em

jogos eletrônicos, onde na Inteligência Artificial, a “busca” se refere a uma técnica de procurar soluções para um problema (WHITBY, 2004).

Segundo Funge (2004), as aplicações de inteligência artificial têm significado diferente quando encontradas em meio acadêmico e IA encontrada em jogos digitais. Para diferenciar uma da outra os desenvolvedores de jogos digitais aderiram o termo Game AI.

A diferença que pode ser vista em IA acadêmica e a IA para jogos digitais é o objetivo que cada uma apresenta. Na IA acadêmica o objetivo é buscar soluções para problemas complexos, desenvolver agentes inteligentes e entender o raciocínio humano (SCHWAB, 2009). Já para jogos digitais a finalidade de se usar uma IA é a diversão para quem está jogando.

Para Tozour (2002), a importância de se usar uma Game AI é para mostrar o resultado de sua implantação, não importando como se chega até este resultado, ou seja, mostrar como o sistema age e não como ele pensa. Isso ocorre pelo fato de que jogos digitais são negociáveis, os consumidores os compram para se divertir, assim não lhes interessando como foi desenvolvido a IA de um agente do jogo, desde que o jogo se torne desafiador e divertido com essa inteligência de forma que o personagem tome decisões que está seguindo com as regras do jogo.

Para que uma IA seja usada em jogos é preciso ponderar nas limitações técnicas bem como o tempo para construir personagens mais ou menos inteligentes. Os oponentes computadorizados não devem ser fáceis de derrotar, mas precisam oferecer um desafio que exija que os jogadores usem habilidades aprendidas no decorrer do jogo para derrotá-los (SCHUYTEMA, 2008).

2.3 A IA E OS JOGOS DIGITAIS

Para entender como a IA se comporta dentro de um jogo Schwab (2004), nos fala que a utilização de uma IA tem como objetivo recriar a inteligência humana e desenvolver suas técnicas de modo a criar agentes inteligentes, deste modo, a IA preocupa-se em como o agente inteligente age e não se preocupa em como ele pensa, seu foco e importância está voltado no resultado, para proporcionar uma jogabilidade melhor e diversão para quem está jogando.

Antes dos personagens nos games possuírem redes neurais, todas as suas instruções eram programadas nos scripts, assim que o jogador alcançasse

determinado objetivo ele viria a fazer o que foi programado, para mudar isso Deloura (2000) apresenta estratégias simplificadas de algoritmos de inteligência artificial voltados para busca, conhecido como A*, este algoritmo aplicado a um jogo digital vai dar a função a um personagem de calcular rotas para encontrar saídas mais rápidas.

Assim, como este algoritmo existem vários outros que dão funções específicas para cada personagem, como exemplo temos a lógica fuzzy apresentada por Tanscheit (2004) onde o personagem apresentará ações diferentes sempre que o jogador interagir com ele.

2.4 TÉCNICAS DE GAME AI UTILIZADAS EM JOGOS DIGITAIS

Existem várias técnicas e algoritmos que são utilizados no processo de criação dos jogos digitais para que os agentes tenham inteligência e personalidade (BOURG, 2004).

Para Lamothe (1999) existem dois tipos básicos de IA encontrado em jogos sendo eles algoritmos determinísticos e algoritmos de padrões de movimento, com esses algoritmos as ações dos agentes são pré-programadas ou processados. Já Dalmau (2004) menciona quatro tipos mais utilizados de IA implantadas em jogos, são elas: algoritmos de pathfinding, máquinas de estados finitos, algoritmos genéticos e sistemas baseados em regras.

2.4.1 MÁQUINAS DE ESTADOS FINITOS

Uma máquina de estados finitos é uma máquina que defini em qual estado um personagem está e pode mudar este estado. O estado atual da máquina defini qual ação ele deve tomar a cada momento. Sua utilização se deve ao fato da técnica ser barata computacionalmente e por ser fácil de entender e fácil de implementar, ela continua sendo utilizada frequentemente nos jogos atuais (BOURG, 2004).

Como exemplo podemos citar um jogo onde um agente inteligente tem as instruções para perseguir o jogador, correr do jogador e ficar parado. Se o agente não ver o jogador pelo mapa ele fica parado, após ele avistar o jogador ele então muda de estado e passa a perseguir o jogador e se o jogador pegar um item em específico o agente muda de estado e passa a correr do jogador, a partir de cada ação tomada pelo jogador o agente ira mudar o seu estado.

2.4.2 ALGORITMOS DE PATHFINDING

As técnicas de algoritmos de pathfinding são utilizadas para resolver problemas de navegação dos personagens, essa técnica também é conhecida como algoritmo de busca. Para Bourg (2004), é importante que um algoritmo de pathfinding seja bem implementado, pois uma busca mal implementada tende a fazer com que os agentes pareçam sem inteligência para se movimentar no mapa e para desviar de barreiras.

Para resolver o problema de pathfinding, Dalmau (2004) cita vários algoritmos que podem ser usados, sendo o mais utilizado termos o A* (A Star), embora outros algoritmos pathfinding como o waypoints e dijkstra são usados. Na maioria dos jogos, os seus desenvolvedores criam um mapa onde o personagem caminha por meio de “grades” (grids), assim cada grade representa um nó de um grafo.

Para Bourg (2004) um problema deste tipo de técnicas é o consumo de processador, quando criamos caminhos pré-calculados, conhecidos como waypoints é possível minimizar o consumo do processador. O waypoints são nós em locais do mapa do jogo que ajudam o agente a se movimentar de um lugar para outro através de rotas pré-calculadas ou métodos de pathfinding com custo baixo.

2.4.3 ALGORITMOS GENÉTICOS

O uso desse algoritmo se dá em jogos que necessitam da geração de uma população, para isso se cria diferentes indivíduos com características únicas, assim podendo ser representado por um valor dentro do vetor. Este tipo de técnica é usado para criação de pessoas em jogos que simulem uma população de uma cidade, esse algoritmo também pode ser usado para criar mutações ou evoluir a espécie de um personagem (DALMAU, 2004).

2.4.4 SISTEMAS BASEADO EM REGRAS

Algumas ações presentes em jogos digitais não são fáceis para se projetar em estado e transição. Para melhor entendimento considere o seguinte exemplo:

- Se há uma fruta por perto e o personagem está com fome, ele irá comer a fruta;
- Se o personagem está com fome, mas não tem nenhuma fruta por perto, ele procura por uma;

- Se o personagem não está com fome, mas está com sono, ele irá dormir;
- Se o personagem não está com fome e não está com sono, ele irá caminhar.

Essas quatro situações não são fáceis de se representar por meio de uma máquina de estados, já que cada sentença vai a um estado de máquina e cada sentença pode mudar para qualquer uma das outras sentenças. Este problema é denominado como comportamento global.

Neste exemplo o personagem tem comportamentos diferentes definido por cada sentença com um conjunto de regras ou prioridades. Esse sistema tem a característica onde uma condição vai gerar uma ação. De acordo com o exemplo a cima as regras seguiriam a seguiriam a forma:

- Fome e fruta por perto gera a ação de comer;
- Fome e sem frutas por perto gera a ação de procurar;
- Sem fome e sem sono gera a ação de dormir;
- Sem fome e sem sono gera a ação de andar;

Através destas condições se verifica se é possível ativar as regras bem como quais ações vão ser tomadas caso a regra seja ativada (DALMAU, 2004).

3 BENEFÍCIOS DE SE USAR UMA INTELIGÊNCIA ARTIFICIAL NOS JOGOS

Para que uma IA traga algum benefício é preciso que ela siga as regras do jogo, que simulem a inteligência e os erros humanos, deve ter personalidade, e devem ter níveis dinâmicos para que assim o jogador tenha a sensação de ser desafiado, pois os jogadores esperam oponentes melhores em jogos complexos. Mas nada disso é válido se o fator principal não for alcançado, a diversão (SCHWAB, 2004).

Nos jogos a IA tende a aumentar a imersão e experiência que o usuário tem, dando um melhoramento em sua jogabilidade. Em jogos para um jogador (single-player) os NPC's (personagens não jogáveis) inteligentes são usados para dar a ilusão que o jogador não está sozinho, e sim em um mundo com vários jogadores inteligentes, além de adicionar uma imersão maior no jogo. Já nos jogos para vários jogadores (multiplayer) a IA pode ser usada a fim de que o jogador não precise que tenha toda uma comunidade atuando dentro do jogo para que ele não se sinta sozinho. Assim como a IA traz benefícios dentro do jogo, ela também traz vantagens na hora de desenvolver o jogo, com um algoritmo de redes neurais o próprio agente vai aprendendo através do erro assim não é necessário programar manualmente todo o seu comportamento (CHAMPANDARD, 2003).

3.1 PROBLEMAS DE INTELIGÊNCIA ARTIFICIAL NOS JOGOS

Os criadores de jogos digitais encontraram várias soluções para jogos através da utilização de uma IA, porém muitos problemas aparecem com o uso de IA (BOURG, 2004). Alguns autores ainda citam fatores que podem ser classificados como maior parte do problema de uma IA em jogos, sendo eles:

Período curto para desenvolvimento: onde o período para se criar um jogo é curto, isso dificulta pois o desenvolvedor não tem tempo suficiente para estudar as tecnologias mais sofisticadas para IA e como aplicá-la em um jogo comercializado (BOURG, 2004).

Algoritmos de aprendizado: este tipo de algoritmo tem os resultados complexos para serem testados, já que não retornam sempre o mesmo resultado. Por isso, diversos desenvolvedores preferem optar por técnicas de IA determinísticas, já que são mais fáceis de implementar, entender e depurar (WOODCOCK, 1999).

Poder de processamento: os jogos são aplicativos que rodam cada frame por segundo, assim suas informações são atualizadas e processadas a cada segundo.

Algoritmos com custo elevado de processamento não devem ser utilizados em jogos que precisam de respostas rápidas (TOZOUR, 2002).

Na criação de um jogo toda a equipe se baseia na documentação feita pelo game design. Mas existe um conflito entre o game AI e o game designers, pois os game designers cria toda a narrativa, definem a jogabilidade e a execução dos eventos do jogo, para que seja possível ter um controle total do jogo e dos NPC's. E então surge o seguinte problema, se os designers definem o comportamento dos NPC's é necessário usar uma IA? Em contra partida, se uma IA estiver apta a dar total autonomia aos NPC's é necessário ter um designer? Para solucionar este problema é necessário que os games designers e os programadores entrem em acordo (CHAMPANDARD, 2003).

4 MOTOR DE JOGOS

Um motor de jogos ou game engine, é um software com um conjunto de bibliotecas e ferramentas para simplificar ou abstrair o desenvolvimento de jogos ou outras aplicações gráficas, otimizando o tempo de produção (COMBOINFINITO.COM, 2019). No mercado atual existem vários motores de jogos como Unreal, Unity, CryEngine, Game Maker entre outros.

Para o desenvolvimento do software proposto para o trabalho de conclusão de curso foi utilizado o motor de jogos unity, pelo mesmo oferecer uma grande variedade de recursos que otimizaram o tempo de desenvolvimento do jogo e por maior familiaridade com o mesmo.

4.1 UNITY

O unity é um motor de jogos que suporta tanto o desenvolvimento de jogos em 3D quanto 2D para satisfazer todas as necessidades exigidas pelo usuário, com esta ferramenta também é possível criar filmes, animações e etc. Com o unity é possível criar jogos de maneira mais pratica, pois o motor de jogos faz toda a parte difícil como a utilização de física que já está integrada em seu sistema, assim o programador não terá o trabalho de criar toda a física para o jogo (UNITY.COM, 2019).

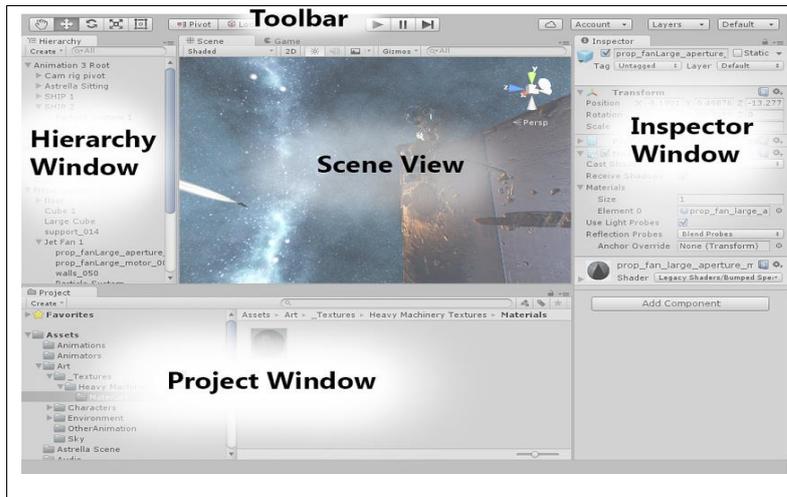
4.2 ESTRUTURA DO UNITY

Para criar um jogo é necessário entender como a ferramenta funciona. Para isso será apresentado alguns conceitos da ferramenta para melhor entendimento.

4.2.2 INTERFACE

A partir da interface podemos ver diversas guias que podem ser organizadas, anexas e encaixadas, tudo conforme as preferencias do desenvolvedor. Na interface encontramos diversos tipos de janelas chamadas de views, cada uma tendo um proposito diferente sendo eles a scene, game, hierarchy, o inspector e project (UNITY.COM, 2019).

FIGURA 1 - Interface do unity



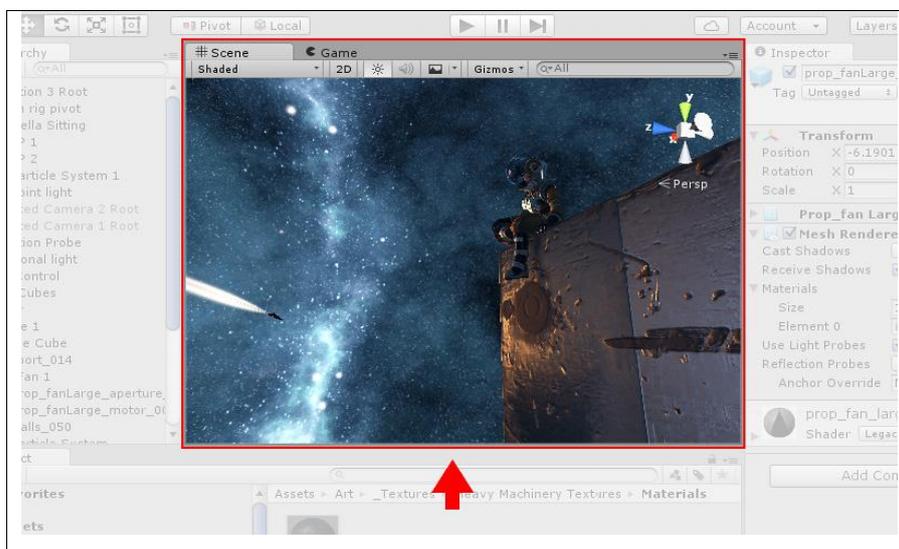
Fonte: Unity.com (2019).

4.2.3 SCENE

A scene view é composta pelos objetos que estão presentes no jogo, sendo eles os Game Objects, a partir da scene view é possível manipular todos esses objetos dentro dela apenas clicando e arrastando com o mouse. Nesta janela é possível ter uma perspectiva do jogo tanto em 3D como em 2D dependendo de como se está trabalhando.

Devido há grande importância que esta janela representa para o desenvolvimento de um projeto, várias formas de navegação são oferecidas para aumentar a produtividade do desenvolvedor (UNITY.COM, 2019).

FIGURA 2 - Scene view

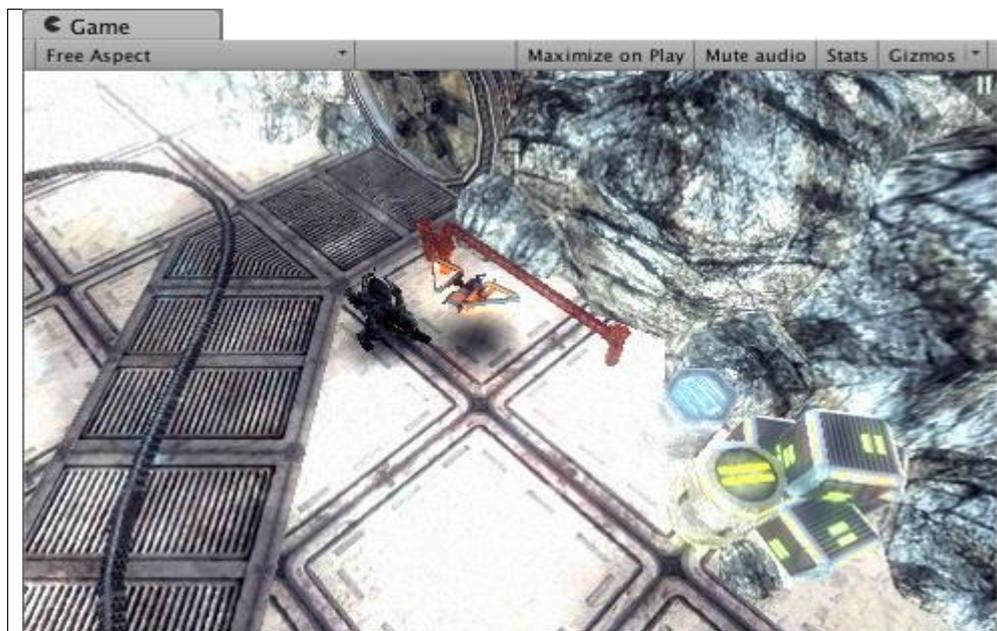


Fonte: Unity.com (2019).

4.2.4 GAME

Nesta janela é possível ter uma visualização da aplicação em desenvolvimento, ou seja, é possível ver como a aplicação estará quando finalizada. A partir dessa tela é possível ter uma prévia de como os elementos do jogo estão se comportando dentro da aplicação. Nesta janela é possível fazer os testes da aplicação para poder detectar erros e bugs, além disso a unity fornece a possibilidade de paralisar a simulação com um botão de pausa enquanto a aplicação estiver em depuração, assim possibilitando que os parâmetros dos elementos possam ser ajustados para experimentação (UNITY.COM, 2019).

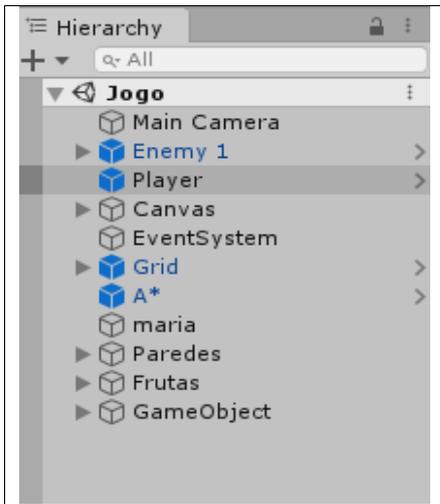
FIGURA 3 - Game view



Fonte: Unity.com (2019).

4.2.5 HIERARCHY

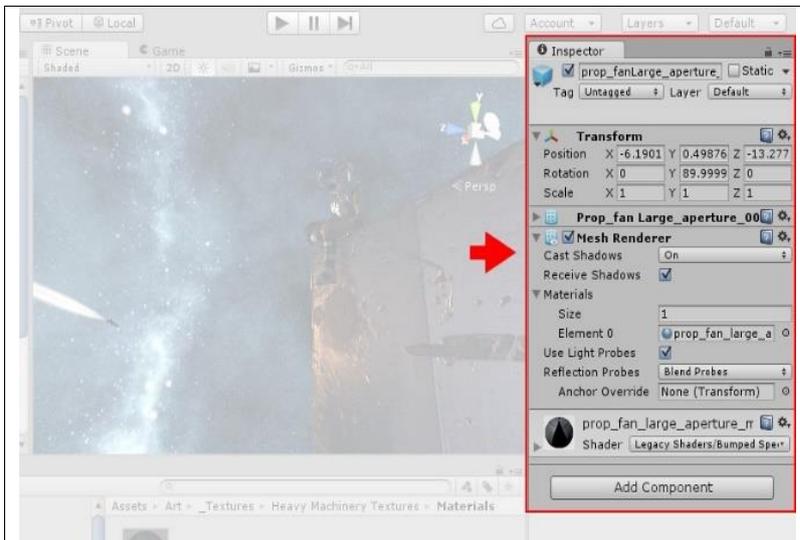
De acordo com o site Unity.com (2019) esta janela exibe todos os elementos de forma hierárquica de texto que se encontram na scene. Cada item que está na scene possui uma entrada na hierarquia, sendo assim as duas janelas vinculadas. Através da hierarquia é possível ver toda a estrutura de como os objetos são anexados um ao outro.

FIGURA 4 - Hierarchy

Fonte: Elaborado pelo Autor (2019).

4.2.6 INSPECTOR

A janela do Inspector nos dá acesso aos vários parâmetros de um objeto presente no cenário e aos atributos de seus componentes. Nesta janela é possível editar todas as propriedades do objeto selecionado, bem como adicionar novas propriedades (UNITY.COM, 2019).

FIGURA 5 - Inspector

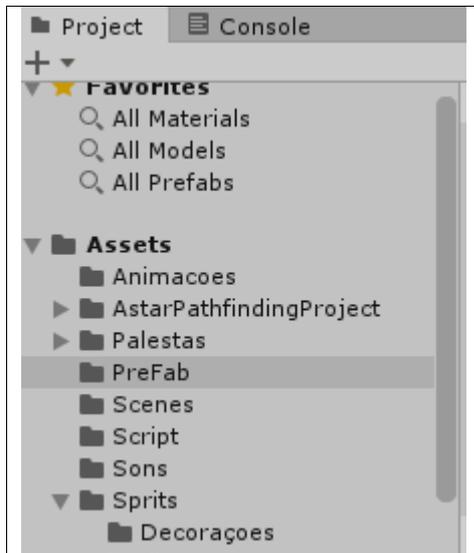
Fonte: Unity.com (2019).

4.2.7 PROJECT

Segundo o site Unity.com (2019) a janela do project é onde se manipula e organiza vários arquivos chamados de assets que compõem um projeto tais como

texturas, prefabs, efeitos de áudio, scripting e etc. A partir desta janela é possível visualizar todas as pastas de assets disponíveis dentro do projeto no sistema de arquivos do computador.

FIGURA 6 - Project



Fonte: Elaborado pelo Autor (2019).

4.2.8 GAME OBJECTS

O Game Object é o conceito mais importante do unity, pois todo objeto presente no jogo é considerado um Game Object, seja ele um personagem, itens coletáveis, luzes, câmeras e efeitos especiais. Um Game Object é apenas um objeto vazio, para que tenha alguma utilidade é necessário adicionar componentes a ele antes que ele se torne algum tipo de elemento para o jogo, a partir de diferentes combinações de componentes adicionados a ele o Game Object assumira uma forma, que pode ser ela um personagem, uma árvore, luzes ou efeitos especiais. Um componente pode ser desde um script, uma geometria de colisão ou uma textura.

No unity temos o conceito de prefabs, um prefab é um modelo de composição de Game Object definido. Como exemplo podemos citar a definição de um humanoide que necessita de um script para se movimentar e um componente de colisão. A partir deste caso, podemos criar um prefab do humanoide e criar várias cópias dele com parâmetros diferentes. Assim o programador terá um ganho de tempo pois evitaria ter que recriar essa composição para cada instância de humanoides presentes no cenário (UNITY.COM, 2019).

4.2.9 ASSETS

Grande parte do desenvolvimento de um jogo está relacionado ao uso e manuseio de assets tais como texturas, sprites, efeitos de som e scripts. Esses assets são desenvolvidos em ferramentas fora do unity, sendo cada ferramenta específica para cada tipo de assets. Após criados estes assets precisam ser importados para dentro do unity, assim o usuário pode importa-los apenas arrastando os assets que deseja para dentro da pasta no project. Após esta ação o asset será importado automaticamente para dentro do unity, possibilitando que o usuário possa manuseá-lo de forma adequada a sua finalidade.

Outra forma de se importa assets para a atual aplicação é através da asset store disponível dentro do unity, através desta janela o usuário é capaz de visualizar, comercializar e utilizar os assets disponíveis pela comunidade, para a atual aplicação assim tirando qualquer outra interação que seria feita com outra ferramenta fora do unity (UNITY.COM, 2019).

4.2.10 SPRITES

Os sprites são objetos gráficos 2D utilizados para criar objetos dentro da scene, diferentes dos objetos 3D que utilizam texturas os sprites 2D utilizam imagens que podem ser editados dentro do unity pois ele fornece uma ferramenta para edição de imagens, a partir dele é possível editar, recorta e modificar a física colocada dentro das sprites (UNITY.COM, 2019).

4.2.11 SCRIPTS

De acordo com o site unity.com (2019) o unity nos dá a opção de desenvolvermos uma aplicação com duas linguagens diferentes sendo elas C# e JavaScript, através de uma destas linguagem criamos os scripts que são essenciais em todos os aplicativos que forem criados no unity. As aplicações criadas no unity requerem a criação de scripts para responder aos comandos de entrada enviado pelo jogador e para organizar os eventos do jogo para que ocorram quando foram programados para acontecer. Os scripts também podem ser utilizados para controlar o comportamento físico dos objetos, criar efeitos gráficos e implementar um sistema de IA personalizado para os personagens do jogo.

4.2.12 SISTEMA DE FÍSICA

Para que um objeto tenha um tipo de comportamento semelhante ao do nosso mundo é necessário que ele possua algum tipo de física, o unity oferece várias formas de usarmos os componentes de física disponíveis em seu sistema, podemos manipular a gravidade e colocar massa corporal em objetos assim tornando-os sólidos.

Existem dois tipos distintos de mecanismos de física no unity um para 3D e outro para 2D, o conceito para os dois são semelhantes exceto que um é para dimensão 3D e outro para dimensão 2D, porém são implementados de formas diferentes. Por exemplo, existe um componente de corpo rígido para física 3D e um Rigidbody 2D análogo para a física 2D (UNITY.COM, 2019).

4.3 C#

De acordo com o site microsoft.com (2019), C# (sharp) é uma linguagem de programação orientada a objetos e fortemente tipada, desenvolvida pela Microsoft e atualmente sendo utilizada na IDE Visual Studio que permite que o desenvolvedor crie uma variedade de aplicativos executados no .NET Framework. Com o C# é possível criar aplicativos de cliente do Windows, serviços Web XML, componentes distribuídos, aplicativos cliente-servidor, aplicativos de banco de dados e outros.

Para se criar scripts para os jogos no unity é necessário o uso de uma IDE, se o usuário estiver usando o unity na versão 2018.1 ou superior, ao instalar o motor de jogos, será instruído ao desenvolvedor instalar o visual studio em sua máquina, pois a IDE já está integrada no ambiente da unity (UNITY.COM, 2019).

4.4 APRESENTAÇÃO DO JOGO FRUIT SHOTING

Como visto nos tópicos anteriores para se criar um jogo no unity é essencial que o desenvolvedor saiba os conceitos básicos e a funcionalidade de cada janela presente no unity. Aplicando os conhecimentos adquiridos foi desenvolvido um jogo com a implementação de uma inteligência artificial usando a técnica de máquina de estados finito.

O fruit shoting é um jogo do gênero de tiro, onde o jogador deve eliminar os inimigos no mapa com disparos. Os inimigos presentes no jogo irão correr atrás do jogador e ficaram a uma certa distância segura para então atirar frutas nele, caso o

jogador tente chegar perto, o inimigo irá recuar para trás, caso o jogador seja atingido pelas frutas disparadas pelo inimigo ele recomeçará na fase onde estava. Caso queira finalizar o jogo basta apertar a tecla esc e então o jogador será levado a tela de fim de jogo onde poderá ser visto o seu placar final, e apertando esc novamente o jogador sairá do software.

O presente jogo conta com uma tela inicial, final e tela de fases, sendo a primeira tela a de entrada do aplicativo com o menu inicial onde o jogador pode iniciar o jogo ou sair do aplicativo, como pode ser visto na figura 7 no apêndice A. Após a tela do menu inicial o usuário irá ver a tela do jogo onde ele poderá interagir controlando o personagem principal usando as setas do teclado para se movimentar e apertando ctrl para atirar nos inimigos, mas o personagem não é capaz de atirar na vertical, a tela de jogo pode ser visualizada na figura 8 no apêndice B. Após derrotar uma certa quantidade de inimigos o jogador será levado para a próxima fase que pode ser visualizada na figura 9 no apêndice C, aqui o jogador deve desviar dos lasers e dos inimigos, pois se ele for pego é fim de jogo. A tela de fim de jogo é onde será mostrado o placar final do jogador como pode ser visto na figura 10 no apêndice D.

4.5 USO DE ASSETS

Para se criar um jogo em 2D é necessário o uso de assets de imagens e sons, para que o desenvolvedor não perca muito tempo criando esses conteúdos quando seu prazo de entrega é curto e o mesmo não possui uma equipe ou habilidades para se criar este tipo de conteúdo. É possível encontrar sites que disponibilizam os assets de forma gratuita e com licença podendo assim ser utilizados nas aplicações, assim como pode-se encontrar assets pagos. Os assets encontrados no jogo fruit shooting estão disponíveis na asset store do unity.

5 CONSIDERAÇÕES FINAIS

A partir das informações e das pesquisas bibliográficas foi analisado que existe uma grande diferença no uso da Inteligência Artificial em jogos e na área acadêmica, pois na área acadêmica o uso de uma IA implica em resolver algum problema, sendo necessário um estudo formal, baseando-se em teorias. Já a IA para jogos está apenas preocupado com o resultado alcançado, já que as técnicas usadas para se chegar ao resultado não são importantes como no estudo acadêmico.

Esta diferença se deve ao fato de que os jogos são produtos comercializados, voltados para o entretenimento, e visto que a indústria de games fatura milhões por ano, por tanto, o prazo para que um jogo seja entregue é curto, assim sendo a IA na área de jogos algo informal.

A fim de atingir o objetivo de demonstrar como a implementação da inteligência artificial (IA) pode aumentar o engajamento dos jogadores em um jogo, foi utilizado algoritmos de IA e com sua implementação os personagens não jogáveis mostraram sinais de inteligência fazendo escolhas sensatas durante o decorrer do jogo e respeitando as regras impostas em cada nível do jogo.

Para demonstrar como uma Inteligência artificial (IA) se comporta dentro de um jogo foi utilizado técnicas de Game AI que foram descritas no capítulo dois, assim os personagens não jogáveis foram capazes de fazer escolha de como agir a cada segundo, respeitando toda a regra do jogo e gerando um nível maior de imersão dentro do jogo.

Como visto no capítulo três a inteligência artificial pode trazer muitos benefícios para um jogo desde que respeite suas regras e que tenha graus de dificuldade a fim de que não seja muito inteligente em níveis iniciais do jogo e que seja mais esperta em níveis mais avançados, pois se a IA aplicada ao personagem for muito inteligente logo no início do jogo é muito provável que ela quebre a imersão do jogador já que ela poderá sempre prever os movimentos do mesmo assim gerando um sentimento de derrota por parte do jogador, já que o mesmo ainda está se familiarizando com as regras impostas pelo jogo. Por outro lado, uma IA que não se torna mais esperta com o decorrer do game pode quebrar a imersão do jogo visto que a mesma não é capaz de fazer uma escolha sensata, assim a IA deixa de proporcionar desafios para o jogador e torna o jogo uma experiência sem graça.

E para que todo o projeto fosse possível foi utilizado o motor de jogos unity que possibilitou toda a criação do jogo, com tudo foi necessário utilizar uma IDE para programar os scripts que fazem toda a lógica do jogo, ao qual foi utilizado o visual studio utilizando a linguagem C#.

A partir das pesquisas bibliográficas e web, a fim de alcançar os objetivos e validar as hipóteses, foi utilizado uma máquina de estados finitos dentro do jogo desenvolvido, a partir dessa técnica de IA foi notado que quando o jogador executa uma ação gera uma reação por parte dos agentes inteligentes, esse tipo de reação causa um impacto gerando maior dificuldades para que o jogador possa prosseguir, porém este impacto não mudou o resultado final do jogo. Com a implementação dessa IA foi notado que é possível equiparar um nível de dificuldade com o decorrer do jogo para que assim haja um engajamento por parte dos jogadores e proporcionando um novo nível de dificuldade e pôr fim a diversão do mesmo.

REFERÊNCIAS

BOURG, David M.; SEEMAN, Glenn. **AI for Game Developers**. Sebastopol: O'Reilly. 2004.

CHAMPANDARD, Alex J. **AI Game Development: Synthetic Creatures with Learning and Reactive Behaviors**. Indianapolis: New Riders. 2003.

COMBOINFINITO. **O que é um motor de jogos e qual sua importância**. Disponível em:<<https://www.comboinfinito.com.br/principal/o-que-e-uma-engine-qual-sua-importancia-no-desenvolvimento-de-games/>>

DALMAU, Daniel Sánchez-Crespo. **Core Techniques and Algorithms in Game Programming**. Indianapolis: New Riders. 2004.

DELOURA, M., **Game Programming Gems (Vol. I)**. Charles River Media Ed..2000. 550 P. ISBN 1584500492.

FUNGE, John David. **Artificial intelligence for computer games: an introduction**. AK Peters/CRC Press, 2004.

HUIZINGA, Johan. **Homo ludens: Proeve fleener bepaling van het spel-element der cultuur [Homo ludens: A study of the play-element in culture]**. Haarlem: Tjeenk Willink, 1938.

IAN, Millington; FUNGE, John David. **Artificial Intelligence for Games**. 2006.

JUUL, J., **Half-Real: Video Games between Real Rules and Fictional Worlds**, The MIT Press, 2005, ISBN: 0262101106.

LAKATOS, Eva Maria; MARCONI, Marina de Andrade. **Fundamentos de Metodologia Científica**. 5. ed. São Paulo: Atlas, 2003.

LAMOTHE, André. **Tricks of the Windows Game Programming Gurus: Fundamentals of 2D And 3D Game Programming**. Indianópolis: Sams. 1999.

LUCCHESI, Fabiano; RIBEIRO, Bruno. **Conceituação de jogos digitais**. São Paulo, 2009.

MICROSOFT. **Introdução a linguagem C#**. Disponível em:<<https://docs.microsoft.com/pt-br/dotnet/csharp/getting-started/introduction-to-the-csharp-language-and-the-net-framework>>

RAMPAZZO, Lino. **Metodologia Científica para alunos dos cursos de graduação e pós-graduação**. 3. ed. São Paulo: Loyola, 2005.

ROGERS, Scott. **Level Up: Um Guia Para o Design de Grandes Jogos**. São Paulo: Blucher, 2010.

RUSSEL, S., **Artificial Intelligence: A Modern Approach**, Prentice Hall, 2a edição, 2002, ISBN-10: 0137903952.

SCHUYTEMA, P. **Design de games: uma abordagem prática**. São Paulo: Cengage Learning, 2008. 447 p.

SCHWAB, Brian. **AI game engine programming**. Nelson Education, 2009.

TANSCHUIT, R; **Sistemas Fuzzy**. DEE-PUC-Rio, Rio de Janeiro (2004).

TOZOUR, Paul. **The Evolution of Game AI from AI Game Programming Wisdom**. Hingham: Charles River Media. 2002.

UNITY. **Unity Manual**. Disponível em: < <https://docs.unity3d.com/Manual> >

WOODCOCK, Steven. **Game AI: The State of the Industry**. Game Developer Magazine. CMP Media. Aug 1999. Pages 34-43.

APÊNDICE A - Tela inicial

FIGURA 7 - Tela inicial.



Fonte: Elaborado pelo autor (2019).

APÊNDICE B - Tela de jogo

FIGURA 8 - Fase 1.



Fonte: Elaborado pelo autor (2019).

APÊNDICE C - Segunda tela de jogo**FIGURA 9 - Fase 2.**

Fonte: Elaborado pelo autor (2019).

APÊNDICE D - Tela de fim de jogo

FIGURA 10 - Tela de fim de jogo.



Fonte: Elaborado pelo autor (2019).