

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS
ESCOLA POLITÉCNICA E DE ARTES
GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO



**CARDBATTLE:
DESENVOLVIMENTO DE UM JOGO DE CARTAS UTILIZANDO ESTRUTURA DE DADOS**

Giuliano Giancarlo Latorraca

Goiânia
2023

GIULIANNIO GIANCARLO LATORRACA

**CARDBATTLE:
DESENVOLVIMENTO DE UM JOGO DE CARTAS UTILIZANDO ESTRUTURA DE DADOS**

Trabalho de Conclusão de Curso apresentado à Escola Politécnica e de Artes, da Pontifícia Universidade Católica de Goiás, como parte dos requisitos para a obtenção do título de Bacharel em Ciência da Computação.

Orientador (a): Fernando Gonçalves Abadia

Goiânia
2023

GIULIANNNO GIANCARLO LATORRACA

**CARDBATTLE:
DESENVOLVIMENTO DE UM JOGO DE CARTAS UTILIZANDO ESTRUTURA DE DADOS**

Este Trabalho de Conclusão de Curso julgado adequado para obtenção o título de Bacharel em Ciência da Computação, e aprovado em sua forma final pela Escola de Ciências Exatas e da Computação, da Pontifícia Universidade Católica de Goiás, em ___/___/_____.

Prof. Ma. Ludmilla Reis Pinheiro dos Santos
Coordenador(a) de Trabalho de Conclusão de Curso

Banca examinadora:

Orientador(a): Prof. Me. Fernando Gonçalves Abadia

Prof. Me. Gustavo Siqueira Vinhal

Prof. Dra. Solange da Silva

Goiânia
2023

RESUMO

O baralho convencional é amplamente conhecido e apreciado por jogadores de todo o mundo. No entanto, suas variações e adaptações têm dado muitas origens a novas abordagens e estratégias ao explorar os baralhos personalizados. Os jogos de cartas com baralhos personalizados são aqueles em que as cartas são pensadas e desenvolvidas exclusivamente para um jogo específico, proporcionando uma experiência única e cativante para os jogadores. O objetivo deste trabalho é o desenvolvimento de um jogo de cartas inovador e envolvente, utilizando um baralho personalizado e a *engine* para o desenvolvimento de jogos, chamada *GameMaker Studio 2*. Além disso, serão aplicadas práticas e conceitos de programação em Estrutura de Dados para otimizar e aprimorar a funcionalidade do jogo. A criação de um baralho personalizado permite uma maior identificação e imersão dos jogadores no universo do jogo, onde cada carta é cuidadosamente projetada, levando em consideração elementos como design, valor numérico para o ataque ou defesa e onde uma carta poderá ser posicionada durante a partida. Essa personalização proporciona uma experiência única e diferenciada, tornando o jogo mais envolvente e estimulante. A *engine GameMaker Studio 2* será utilizada como uma ferramenta para o desenvolvimento do jogo, onde a mesma oferece recursos avançados e uma interface intuitiva, permitindo a criação de mecânicas de jogo complexas e interativas. Através desse motor gráfico, será possível implementar as regras do jogo, gerenciar as interações entre as cartas e criar uma experiência visualmente atraente para os jogadores. A aplicação de práticas de programação em Estrutura de Dados irá desempenhar um grande papel fundamental no desenvolvimento do jogo. Serão utilizadas estruturas de pilhas para gerenciar a ordem das cartas e as interações entre elas. Além disso, técnicas de embaralhamento e aleatoriedade serão aplicadas para garantir a imprevisibilidade e o desafio estratégico do jogo. Ao final do projeto, espera-se obter um jogo de cartas com um baralho personalizado que seja divertido, estimulante e capaz de atrair uma variedade de jogadores. A criação desse jogo contribuirá não apenas para o desenvolvimento de habilidades técnicas e criativas, mas também para a expansão do conhecimento na área de desenvolvimento de jogos.

Palavras-chave: baralho personalizado, jogo de cartas, *GameMaker Studio 2*, Estrutura de Dados, desenvolvimento de jogos.

ABSTRACT

The conventional deck of cards is widely known and appreciated by players around the world. However, its variations and adaptations have given rise to new approaches and strategies when exploring custom decks. Card games with custom decks are those in which the cards are designed and developed exclusively for a specific game, providing a unique and captivating experience for players. The objective of this work is the development of an innovative and engaging card game using a custom deck and the game development engine called GameMaker Studio 2. Additionally, programming practices and concepts in Data Structures will be applied to optimize and enhance the game's functionality. Creating a custom deck allows for greater identification and immersion of players in the game universe, where each card is carefully designed, taking into consideration elements such as design, numeric value for attack or defense, and where a card can be positioned during the game. This customization provides a unique and differentiated experience, making the game more immersive and stimulating. The GameMaker Studio 2 engine will be used as a tool for game development, offering advanced features and an intuitive interface, enabling the creation of complex and interactive gameplay mechanics. Through this graphics engine, it will be possible to implement the game rules, manage interactions between cards, and create a visually appealing experience for players. The application of programming practices in Data Structures will play a crucial role in the game's development. Stack structures will be used to manage the order of cards and interactions between them. Additionally, shuffling and randomness techniques will be applied to ensure unpredictability and strategic challenge in the game. At the end of the project, it is expected to obtain a card game with a custom deck that is fun, stimulating, and capable of attracting a variety of players. The creation of this game will contribute not only to the development of technical and creative skills but also to the expansion of knowledge in the game development field.

Keywords: custom deck, card game, GameMaker Studio 2, Data Structures, game development.

LISTA DE FIGURAS

Figura 1 - Jogo da memória com as cartas viradas.	13
Figura 2 - As cartas do baralho do jogo Uno.	14
Figura 3 - A diversidade de cartas do Magic the Gathering sendo vendidas.	15
Figura 4 - O campo de batalha do Magic the Gathering.	16
Figura 5 - Cartas do jogo Red7.	17
Figura 6 - Cartas e fichas de pontuação do jogo Lhama.	18
Figura 7 - Demonstração de como são as peças do jogo Dixit.	19
Figura 8 - Exemplo de diferença de força entre as cartas.	20
Figura 9 - Baralhos e Campo de combate.	21
Figura 10 - Ataque e ganho de pontos.	21
Figura 11 - Exemplo da interface de produção do motor gráfico do Unity.	22
Figura 12 - Exemplo da interface de produção do motor gráfico da Unreal Engine.	23
Figura 13 - Exemplo da interface de produção do motor gráfico do Godot.	24
Figura 14 - Exemplo da interface de produção do motor gráfico do Game Maker 2.	26
Figura 15 - Empilhando e desempilhando dados.	27
Figura 16 - Ilustração da definição de um elemento em uma grade.	28
Figura 17 - BrainStorm.	30
Figura 18 - Ideia de regras e funcionalidades.	32
Figura 19 - Fluxograma do jogo Card Battle.	33
Figura 20 - Esboço do teste do jogo CardBattle.	35
Figura 21 - Ilustração do campo de combate do jogo CardBattle.	36
Figura 22 - Ilustração da carta de Monstro.	37
Figura 23 - Ilustração da carta de Monstro atacando.	37
Figura 24 - Ilustração da carta de Monstro atacando.	38
Figura 25 - Ilustração da carta de Monstro atacando.	38
Figura 26 - Sprites das cartas.	39
Figura 27 - Sprite do campo de batalha.	39
Figura 28 - Objetos utilizados.	40
Figura 29 - Menu.	41
Figura 30 - Baralho.	42
Figura 31 - Campo de Combate.	44

SUMÁRIO

1. INTRODUÇÃO	8
1.1 OBJETIVOS GERAIS	9
1.2 OBJETIVOS ESPECÍFICOS	9
1.3 METODOLOGIA	9
1.4 ORGANIZAÇÃO DO PROJETO	9
2. FUNDAMENTAÇÃO TEÓRICA	11
2.1 A HISTÓRIA DO BARALHO	11
2.2 JOGOS COM BARALHO PERSONALIZADO	12
2.2.1 Jogos de Memória	13
2.2.2 Uno	13
2.2.3 Magic the Gathering	14
2.2.4 Red7	16
2.2.5 Lhama	17
2.2.6 Dixit	18
3. MATERIAIS E MÉTODOS	20
3.1 CARDBATTLE	20
3.2 MOTOR GRÁFICO	21
3.2.1 UNITY	21
3.2.2 UNREAL ENGINE	22
3.2.3 GODOT	24
3.2.4 GAME MAKER STUDIO 2	25
3.3 ESTRUTURA DE DADOS	26
3.4 TESTES	28
4. DESENVOLVIMENTO DO PROJETO	29
4.1 DESENVOLVIMENTO DO PROJETO DE BAIXA FIDELIDADE	29
4.2 O PROCESSO DE DESIGNER	36
4.3 TRANSIÇÃO PARA O GAMEMAKER 2	39
5. RESULTADOS	41
5.1 Tela Inicial e Menu	41
5.2 Jogadores	41
5.3 Baralho de Cartas	42
5.4 Campo de Combate	42
5.5 Pontuação e Ataques	43
5.6 Condições de Vitória	43
6. CONSIDERAÇÕES FINAIS	44
REFERÊNCIAS BIBLIOGRÁFICAS	45

1. INTRODUÇÃO

A evolução dos jogos de baralho ao longo dos anos tem chamado a atenção de pessoas de diversas nacionalidades e faixas etárias, rompendo com a antiga concepção de que esses jogos eram meramente baseados em sorte (Gurney, 1931). Com o avanço contínuo da tecnologia, surgiram no mercado virtual novos jogos de baralho digital que facilitam a jogabilidade sem a necessidade de um grupo pré-existente para participar (Ludopedia, 2023).

Paralelamente, os motores gráficos utilizados no desenvolvimento de jogos eletrônicos têm crescido significativamente, proporcionando uma ampla gama de ferramentas disponíveis no mercado de criação de jogos e esse crescimento permitiu que pequenos desenvolvedores, conhecidos como *Indies*, adentrassem o cenário compartilhando espaço com as grandes produtoras de jogos (Rogers, 2014). Os jogos *Indies* são desenvolvidos por produtoras independentes, que geralmente possuem equipes reduzidas e recursos financeiros limitados (Rogers, 2014). Nesse contexto, muitas vezes é necessário optar por ferramentas de desenvolvimento mais acessíveis, sejam gratuitas ou de custo reduzido. Entre os motores gráficos utilizados para o desenvolvimento de jogos, destacam-se opções como *Unity (Unity Engine, 2023)*, *Unreal Engine (Unreal, 2023)*, *Godot (Godot Engine, 2023)* e *Game Maker Studio 2 (GameMaker, 2023)*, cada qual com suas características e níveis de complexidade.

Nesse contexto, o presente trabalho tem como objetivo principal demonstrar o desenvolvimento de um jogo de combate de cartas personalizadas independente, explorando a aplicação da Estrutura de Dados para otimizar a mecânica e as funcionalidades do projeto. A estrutura de dados consiste em uma área da ciência da computação que trata do armazenamento, organização e manipulação eficiente dos dados em um programa ou sistema. Sua aplicação adequada pode contribuir significativamente para o desempenho e a eficiência de um jogo, permitindo o gerenciamento e o processamento rápido das informações (Cormen, 2012).

Além disso, serão investigadas pesquisas e estudos relacionados ao desenvolvimento de jogos de cartas, tanto no aspecto prático quanto teórico. Serão examinadas abordagens utilizadas por desenvolvedores independentes para a criação de jogos de sucesso nesse gênero, bem como teorias e conceitos relacionados à experiência do jogador, equilíbrio de *gameplay* e *design* de interface (Gil, 2017; Rogers, 2014).

Dessa forma, surge a problemática de compreender a melhor forma de aplicar as técnicas e processos relacionados à Estrutura de Dados no contexto específico do desenvolvimento de um jogo de combate de cartas independente. Para responder a essa questão, é necessário explorar os conhecimentos adquiridos no estudo da Estrutura de Dados e identificar as abordagens mais eficientes para implementar os elementos essenciais do jogo, como a organização das cartas, as interações entre os jogadores e as regras do combate.

Considerando a importância do escopo desse projeto, é fundamental direcionar a programação das aplicações de forma que a Estrutura de Dados seja integrada de maneira eficiente às funcionalidades e mecânicas do jogo. A correta utilização da Estrutura de Dados contribuirá para a otimização do código, o gerenciamento dos recursos do jogo e proporcionará uma experiência fluida e imersiva aos jogadores (Cormen, 2012).

Por fim, este trabalho busca contribuir para a compreensão da aplicação da Estrutura de Dados no desenvolvimento de jogos de combate de cartas personalizadas independentes e

espera-se que os resultados obtidos possam servir como referência e orientação para futuros desenvolvedores que desejam criar jogos nesse gênero, proporcionando-lhes uma base sólida de conhecimento teórico e prático.

1.1 OBJETIVOS GERAIS

- O desenvolvimento de um jogo de cartas utilizando um baralho personalizado e a aplicação da Estrutura de Dados da melhor forma possível.

1.2 OBJETIVOS ESPECÍFICOS

- Mostrar o estudo, a identificação e as técnicas de Estrutura de dados que foram utilizadas no desenvolvimento do jogo de combate de cartas.
- Salientar as tecnologias e as ferramentas que foram utilizadas durante o desenvolvimento.
- Demonstrar os testes e as possíveis falhas encontradas.
- Explicitar como foi o desenvolvimento do jogo de combate de cartas.

1.3 METODOLOGIA

A metodologia que será utilizada em pauta, será a de pesquisa bibliográfica do tipo descritiva, retratando as características do item estudado, expressando com total nitidez os acontecimentos e demonstrando como ele pode ser aplicado dentro do tema proposto.

1.4 ORGANIZAÇÃO DO PROJETO

A estrutura desta documentação é composta por diferentes etapas, cada uma delas desempenhando um papel essencial no desenvolvimento do projeto.

O primeiro ciclo é a Fundamentação Teórica, que se propõe a explorar e apresentar uma explicação mais aprofundada sobre a história e a evolução do baralho ao longo do tempo. Nessa seção, serão abordadas informações relevantes sobre a origem do baralho, suas variações culturais e os diferentes usos que ele adquiriu ao longo dos anos. Além disso, serão discutidos estudos e pesquisas realizados por outros autores que se debruçaram sobre o tema, oferecendo uma visão ampla e embasada sobre o assunto.

O segundo ciclo é dedicado à seção de Materiais e Métodos, onde serão apresentados detalhes específicos sobre o tema proposto neste projeto. Será explicado em maior profundidade qual é o motor gráfico selecionado para a criação do jogo de combate de cartas independente, bem como as técnicas e abordagens utilizadas no processo de desenvolvimento. Será destacado o uso de ferramentas acessíveis e econômicas, com o intuito de possibilitar a participação de pequenos desenvolvedores independentes nesse mercado competitivo. Será fornecido um panorama geral sobre as etapas e processos envolvidos, oferecendo aos leitores uma compreensão completa do trabalho realizado.

O terceiro ciclo, denominado Resultados, é destinado a explicitar as informações e descobertas obtidas ao final do projeto. Serão apresentados os resultados alcançados, verificando se as metas e objetivos estabelecidos foram alcançados com sucesso. Essa seção servirá como uma avaliação do trabalho desenvolvido, proporcionando uma visão clara do que foi conquistado e permitindo uma análise crítica dos resultados obtidos.

Por fim, na seção de Conclusão, será realizada uma síntese de todo o projeto, avaliando se o resultado desejado foi alcançado. Além disso, serão discutidas possíveis melhorias e aprimoramentos para trabalhos futuros, destacando áreas que poderiam ser exploradas ou aperfeiçoadas. Essa parte final servirá como uma reflexão sobre todo o processo de desenvolvimento e contribuirá para o entendimento das implicações e do potencial do jogo de combate de cartas independente no contexto dos jogos digitais.

2. FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão demonstrados os fundamentos teóricos realizados durante o período de pesquisa e que foram aplicados na elaboração deste trabalho.

2.1 A HISTÓRIA DO BARALHO

Há mais de 600 anos, o baralho tem passado por inúmeras transformações ao longo dos séculos, atravessando diferentes países e culturas. Sua origem remonta a fábulas, comentários, sátiras e cenas da vida cotidiana, até chegar à forma de baralho que conhecemos hoje. As cartas de baralho podem ser classificadas em três categorias principais: adivinhação, jogos de azar e uso educativo ou recreativo (Gurney, 1931).

Embora a origem exata das cartas ainda seja incerta para estudiosos, muitas especulações indicam que elas possivelmente tiveram origem na China durante a dinastia Tang, no século IX d.C. Ao longo do tempo, o baralho teria se espalhado para o Ocidente por meio de ciganos, cruzados e comerciantes. Embora não haja provas concretas dessas teorias, é inegável que o baralho tem desempenhado um papel significativo no entretenimento de várias culturas ao longo dos anos (Gurney, 1931).

Cada região ou cultura desenvolveu seu próprio estilo de baralho, baseado em seus costumes e eventos, revelando ao mundo suas práticas e riquezas. Apesar das diferenças, a criação das cartas e dos baralhos manteve certa uniformidade. Alguns baralhos apresentavam reis representados de maneira extravagante, sentados em cavalos ou em tronos. Outros utilizavam imagens do cotidiano, como animais, plantas, pássaros e flores. Esses padrões únicos contribuíram para a evolução das cartas e do baralho ao longo do tempo, transformando-os em uma forma de comunicação singular (Gurney, 1931).

O baralho que conhecemos hoje possui semelhanças marcantes com os primeiros baralhos europeus do século XV, que eram feitos de papelão e tinham ilustrações feitas à mão. Esse baralho evoluiu para o formato atual, composto por um total de 52 cartas, divididas em quatro naipes que simbolizam as estações do ano: ouro (primavera), paus (verão), copas (outono) e espadas (inverno). Cada naipe contém treze cartas, representando os treze dias de cada estação. Ao somar o valor das treze cartas e multiplicá-lo pelo número de naipes, obtém-se o valor de 364. Além disso, há duas cartas coringas, que foram adotadas dos baralhos de Tarô, originalmente utilizados para consulta divinatória. A presença das cartas coringas totaliza 365, representando os dias do ano, e com a segunda carta coringa, o total é de 366, simbolizando o ano bissexto (Gurney, 1931).

Outro aspecto interessante é a relação entre o nome de cada carta do baralho em diferentes idiomas. Ao soletrar o nome de cada carta em inglês ou em três outros idiomas e então realizar a soma do total de letras que cada palavra possui, chega-se ao número 52 que é o total exato de cartas existentes no baralho convencional (*ACE TWO THREE FOUR FIVE SIX SEVEN EIGHT NINE TEN JACK QUEEN KING*) (Gurney, 1931).

Com o passar do tempo, novos tipos de cartas e baralhos foram surgindo, cada um com suas próprias características e peculiaridades. Esses baralhos personalizados foram criados para atender a diferentes interesses, como *hobbies*, colecionismo, temáticas específicas ou até mesmo como forma de expressão artística.

Os baralhos personalizados oferecem uma variedade de designs e estilos únicos, que podem ser baseados em filmes, séries, livros, jogos, culturas, arte, entre outros temas. Além disso, muitos artistas e ilustradores têm se dedicado a criar baralhos personalizados, explorando sua criatividade e talento para produzir conjuntos de cartas verdadeiramente únicos e cativantes (Ludopedia, 2023).

Esses novos tipos de baralhos têm conquistado um público apaixonado e colecionadores ao redor do mundo. Eles não apenas adicionam uma dimensão visualmente atraente ao jogo, mas também proporcionam uma experiência temática mais imersiva, permitindo que os jogadores se conectem de forma mais profunda com seus interesses e paixões (Ludopedia, 2023).

Além disso, a popularidade dos baralhos personalizados se estende além dos jogos tradicionais de cartas. Eles também são usados em truques de mágica, apresentações de cartomagia e até mesmo como objetos de decoração. Com seu apelo estético e a capacidade de contar histórias por meio das ilustrações, os baralhos personalizados se tornaram uma forma de arte em si mesma (Ludopedia, 2023).

O estudo dos baralhos personalizados pode ser abordado de diversas perspectivas, como a análise de seus elementos visuais, a compreensão de seu impacto cultural e social, ou até mesmo a investigação de sua evolução ao longo do tempo (Ludopedia, 2023; Gurney 1931).

Assim, é evidente que o baralho, ao longo dos séculos, transcende seu papel original como uma simples ferramenta de jogo. Ele se tornou um objeto de expressão criativa, representação cultural e entretenimento, desempenhando um papel significativo nas sociedades e deixando um legado rico e diversificado para ser explorado e apreciado. Nesse sentido, este trabalho se propõe a desenvolver um jogo eletrônico com estes tipos de baralhos, com a intenção de aplicar a programação da Estrutura de Dados da melhor forma possível.

2.2 JOGOS COM BARALHO PERSONALIZADO

Diferente do baralho tradicional, o baralho personalizado muitas das vezes tinha um foco mais colecionável, onde cada pessoa deveria adquirir as suas próprias cartas e assim, montar o seu baralho de acordo com as suas estratégias de jogo para conseguir se sobressair contra o seu adversário (Ludopedia, 2023).

Alguns jogos com baralho personalizado variam desde simples jogos de memória a jogos de combate com um sistema muito complexo. A seguir, serão demonstrados exemplos destes jogos:

- *Jogos de Memória*
- *Uno*
- *Magic the Gathering*
- *Red7*
- *Lhama*
- *Dixit*

2.2.1 Jogos de Memória

O *jogo da memória*, de acordo com William Gurney (1931), é um jogo bem antigo, que surgiu na China no século XV, onde o jogador deveria colocar em prática as suas habilidades de foco, dedução e atenção. A partida consiste em várias peças ou cartas viradas para o jogador, onde existem desenhos ou símbolos repetidos, o participante deverá memorizar onde estão as peças que se repetem, depois de um tempo, todas as peças são viradas e o jogo começa.

O jogador deve virar duas cartas que ele acredita serem iguais. Caso sejam diferentes, ele deve colocar as cartas viradas em seu respectivo lugar e tentar novamente. Caso as cartas sejam iguais, ele continua jogando até conseguir acertar todas as peças. Um exemplo do jogo pode ser visto na Figura 1.

Figura 1 - *Jogo da memória* com as cartas viradas.



Fonte: *Jogo Rei da Memória*, app Google Play, 2015.

2.2.2 Uno

Uno, é um jogo de cartas bastante popular entre os jovens e as famílias. Foi criado em 1971 por Merle Robbins para tentar resolver uma discussão com o seu filho sobre as regras de um jogo que se chamava *Crazy Eights*. O jogo acabou sendo um sucesso até que ele percebeu que poderia vendê-lo e fazer com que mais pessoas o jogassem, mais tarde vendendo os direitos para a *International Games* e *royalties*. O jogo evoluiu com o passar dos anos de diferentes formas e até hoje é muito jogado. O *Uno* possui um baralho formado por um total de 114 cartas. Em vez dos quatro naipes, o jogo possui quatro cores: vermelho, amarelo, verde e azul.

Cada cor possui uma marcação de 1 a 9 e mais 5 tipos de cartas especiais. A carta *pular*, onde o próximo jogador perde a sua vez, a carta *reverter*, que faz com que a ordem de jogadas da partida se inverte para o lado contrário, a carta *comprar duas cartas*, onde o próximo jogador deverá comprar mais duas cartas do baralho, a *carta coringa* que ao ser jogada o jogador escolhe qual cor deve ser usada na próxima jogada e a *carta coringa comprar quatro cartas*, onde o próximo jogador deverá comprar mais quatro cartas.

Os jogadores devem combinar a cor ou o número das cartas até que tenham apenas uma carta na mão e dizer a palavra *Uno*. Quando o jogador conseguir jogar a última carta e

não tiver mais nenhuma carta em suas mãos ele vence (Ludopedia, 2014). Um exemplo das cartas do jogo pode ser visto na Figura 2.

Figura 2 - As cartas do baralho do jogo *Uno*.



Fonte: Ludopedia, 2014.

2.2.3 Magic the Gathering

Magic the Gathering é um dos primeiros jogos de baralho de cartas personalizáveis criado oficialmente pelo professor universitário de Matemática, Richard Garfield em 1993. É considerado o jogo de cartas mais complexo já feito e para alguns, é visto como um jogo social. *Magic* é um jogo que envolve um grande e vasto sistema de estratégia, raciocínio e lógica e que também juntou a euforia que muitas pessoas tinham na época em querer colecionar diferentes coisas.

As cartas do jogo Magic são divididas em comuns, incomuns e raras. Ninguém sabia exatamente quantas cartas existiam e quais eram. Isso aumentava a vontade de todos de se interessarem pelo jogo e desejar colecionar todas as cartas. No entanto, era muito difícil, pois as cartas vinham em pacotinhos fechados e não era possível saber quais cartas vinham dentro. Todos tinham que contar com a sorte (Ludopedia, 2014). Um exemplo da diversidade de cartas pode ser visto na Figura 3.

Figura 3 - A diversidade de cartas do *Magic the Gathering* sendo vendidas.



Fonte: *Ludopedia*, 2014.

O contexto do jogo se baseia em um grande duelo entre magos muito poderosos. Eles tinham a capacidade de viajar entre diversos tipos de universos e dominar as magias e os segredos que ali existiam. Basicamente, você é o mago. Quanto mais cartas você possuísse, significava que o seu mago é o mais viajado entre os mundos. Logo, você seria o mais poderoso por saber dominar o maior número de feitiços.

O baralho é dividido em cinco tipos ou cores, onde cada cor determina como será a sua estratégia ou filosofia contra o seu adversário, você irá jogar de forma bruta? Ou de forma astuta? Ou talvez você irá jogar sujo? Ou quem sabe de uma forma rápida? O jogo possibilita todas estas estratégias e até mesmo novas formas de jogo podem ser criadas.

Normalmente o baralho de Magic possui um total de 60 cartas, podendo conter até menos ou mais, isso varia da estratégia de cada jogador. O jogo deve começar com as cartas terreno, que são as cartas que precisam ser jogadas primeiro e que te darão a possibilidade de conjurar monstros ou magias daquele tipo de terreno ou mundo que foi criado, logo, o baralho precisa ser montado de forma inteligente, pois se você tiver terrenos demais ou terrenos de menos, você irá perder o jogo. Durante a partida as suas cartas podem interagir, com os terrenos, com os terrenos do adversário, com os seus monstros, com os monstros do adversário, com os próprios jogadores e até com cartas que você ainda irá jogar. Vence aquele que conseguir passar ou destruir os monstros do adversário e atacar diretamente os pontos de vida dele até chegar a zero (Ludopedia, 2014). Um exemplo de como é o jogo pode ser visto na figura 4.

Figura 4 - O campo de batalha do *Magic the Gathering*.



Fonte: Ludopedia, 2014.

2.2.4 Red7

Red7 é um jogo de cartas criado por Carl Chudyk, onde o objetivo é estar sempre vencendo o máximo de rodas possíveis durante a partida. O jogo se inicia com o embaralhamento do baralho e logo depois, cada jogador receberá um total de 7 cartas e mais uma carta de informação, que demonstra quais cores são as mais fortes, para que ele tenha uma noção de qual carta deveria jogar. No centro da mesa é colocado a carta de regra inicial, que é a *carta mais alta vence* e então a partida começa sempre no sentido horário. Cada jogador receberá uma carta adicional já virada na mesa e assim será determinado quem será o jogador inicial pela força de sua carta.

As cartas são divididas em 7 cores que representam todas as cores do arco-íris e cada cor possui as suas próprias regras quando jogadas no meio da mesa, além das cores o baralho possui 7 tipos de coleções cartas que vão de 1 a 7. A força das cartas varia de acordo com o maior número que possuem e principalmente a sua cor. A cor vermelha (regra: a carta mais alta vence) é a mais forte de todas, seguido da cor laranja (regra: mais do mesmo número vence), amarela (regra: mais de uma mesma cor vence), verde (regra: mais números pares vence), azul (regra: mais cores diferentes vence), anil (regra: mais números em sequência vence) e violeta (regra: mais números abaixo de 4 vence).

Cada jogador possui 3 tipos de ações que podem ser realizadas durante o seu turno. Ele pode escolher uma carta da sua mão e jogá-la. Ele pode tentar jogar uma carta maior do que as outras cartas e tentar vencer a rodada, ou ele pode jogar uma carta por cima da regra inicial, mudando a regra do jogo a seu favor, ou ele pode jogar uma carta na mesa e depois outra carta por cima da regra inicial, mudando a regra do jogo a seu favor. Caso o jogador não conseguir jogar nenhuma carta, então ele estará fora da rodada até que sobre apenas um único jogador, sendo assim o vencedor (Ludopedia, 2017). Um exemplo de como são as cartas do jogo pode ser visto na Figura 5.

Figura 5 - Cartas do jogo *Red7*.



Fonte: *Ludopedia*, 2017.

2.2.5 *Lhama*

Lhama é um jogo de cartas criado por Reiner Knizia que pode ser jogado de 2 até 6 jogadores em partidas de turnos alternados no sentido horário e o objetivo do jogo é ter o menor número de pontos quando a pontuação de alguém durante a partida chegar a 40 pontos.

O jogo se inicia com o embaralhamento do baralho de cartas e neste baralho é possível ter números que vão de 1 a 6 e cartas chamadas de *Lhama*. Os pontos são contados através de fichas de cor branca que correspondem a um ponto e as fichas de cor preta que correspondem a dez pontos.

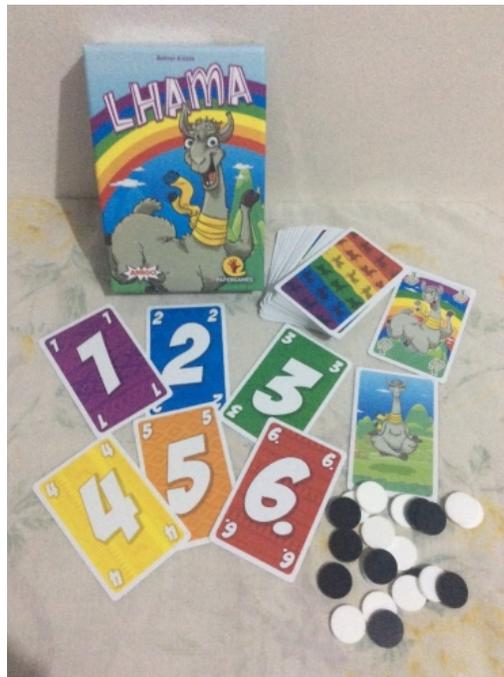
Cada jogador receberá um total de 6 cartas e logo depois a carta no topo do baralho será descartada e jogada no centro da mesa. Cada jogador pode realizar até 3 ações. Uma delas é, o jogador da vez, deve jogar uma carta com o número maior que a carta que foi descartada sobre a mesa e caso a maior carta na mesa seja o número 6, o jogador da vez poderá jogar uma carta de número 6 ou uma carta *Lhama*. Caso a *Lhama* for a carta sobre a mesa, o jogador da vez poderá jogar outra carta *Lhama* ou recomeçar a contagem jogando a carta de número 1.

A segunda ação escolhida pelo jogador, pode ser, comprar uma carta no topo do baralho e esperar a sua vez de jogar. A terceira ação possível, seria abandonar a partida jogando todas as cartas sobre a mesa viradas para baixo e esperar pela próxima rodada.

A rodada termina quando um jogador tiver jogado a sua última carta ou quando todos os jogadores tiverem abandonado a partida e assim que a rodada se encerrar é feito a pontuação.

Cada carta não jogada será convertida em pontos desde que não sejam cartas iguais, os pontos concedidos a cada jogador irá variar de acordo com a soma dos números em suas cartas, logo, o jogo se inicia novamente com as regras iniciais. A partida termina assim que 1 jogador atingir 40 pontos e aquele que tiver o menor número de pontos, vence (*Ludopedia*, 2020). Um exemplo de como são as cartas do jogo pode ser visto na Figura 6.

Figura 6 - Cartas e fichas de pontuação do jogo *Lhama*.



Fonte: Ludopedia, 2020.

2.2.6 Dixit

Dixit é um jogo de dedução e adivinhação criado por Jean-Louis Roubira que pode ser jogado em até 8 pessoas.

O início do jogo se dá, com cada jogador escolhendo a sua cor, um número de cartelas correspondendo ao tanto de jogadores que irão jogar a partida e um total 6 cartas para cada jogador após o embaralhamento do baralho.

A partida começa com um jogador pegando as suas cartas e escolhendo uma delas sem mostrar para os outros jogadores, então ele deve dizer alguma pista sobre o que poderia ser a carta em sua mão para que os outros jogadores tentem adivinhar qual carta é. A ideia é dar alguma dica que não faça com que todos os outros jogadores acertem a carta facilmente. Logo, os outros jogadores devem pegar as suas cartas sem que os outros vejam e escolher uma das cartas que tenha haver com a dica que foi dada. Quem deu a dica deve pegar todas as cartas escolhidas, embaralhar e colocá-las na mesa para que os outros jogadores tentem descobrir qual é a sua carta. A votação é feita com as cartelas que foram dadas no início da partida e então é feita a pontuação de cada jogador. Após isso, as cartas que foram jogadas são descartadas e cada jogador receberá mais uma carta do topo do baralho e a devolução de suas cartelas. O jogo se encerra no momento em que a última carta da pilha do baralho é comprada.

A pontuação é feita da seguinte forma, caso todos os jogadores acertem qual carta da pessoa de quem deu a dica, todos os jogadores que votaram, ganham 2 pontos e andam duas casas no tabuleiro. Caso ninguém vote na carta da pessoa de quem deu a dica, todos os jogadores que votaram também ganham 2 pontos e andam duas casas no tabuleiro. Caso a

carta da pessoa que deu a dica tiver pelo menos 1 voto, a pessoa que deu a dica e quem votou, ganham 3 pontos e anda 3 casas no tabuleiro, cada outro jogador que teve votos em sua carta, ganha 1 ponto adicional, (Ludopedia, 2013). Um exemplo de como é a composição das peças do jogo pode ser visto na Figura 7.

Figura 7 - Demonstração de como são as peças do jogo *Dixit*.



Fonte: Ludopedia, 2013.

3. MATERIAIS E MÉTODOS

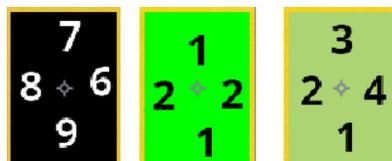
Neste capítulo, serão abordadas todas as fases e programas utilizados no desenvolvimento de um jogo de combate de cartas, com base nos conhecimentos em Estrutura de Dados adquiridos durante o curso de Ciência da Computação na Pontifícia Universidade Católica de Goiás.

3.1 CARDBATTLE

CardBattle é um jogo eletrônico de combate de cartas personalizadas onde o objetivo é conseguir o maior número de pontos enquanto ataca e derrota as cartas do oponente.

Cada jogador possui o seu próprio baralho com um total de 30 cartas que foi montado de forma aleatória e seguindo certas regras. O baralho pode conter no máximo uma única carta considerada lendária e 29 cartas devem ser todas monstros de ataque, onde as cartas devem seguir uma sequência de poder que vai de 1 a 9. Um exemplo de diferença entre as forças das cartas pode ser visto na Figura 8.

Figura 8 - Exemplo de diferença de força entre as cartas.

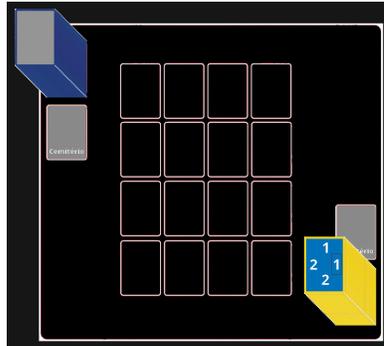


Fonte: *Elaborado pelo autor: Giuliani, 2023.*

O jogo possui um único modo de dificuldade contra a máquina que o jogador terá que desafiar, o modo normal. A máquina às vezes irá tentar derrotar as suas cartas ou ela jogará a próxima carta de forma aleatória.

O jogo se inicia com o embaralhamento dos baralhos e o jogador apenas consegue saber qual é a carta do topo do seu baralho, enquanto as cartas da inteligência artificial ficarão escondidas de seus olhos. O iniciador da jogada é escolhido de forma aleatória que deve pegar uma carta no topo do seu baralho e decidir onde ele deverá jogar esta carta no campo de combate. Cada jogador só pode ter uma carta em suas mãos e então ele é obrigado a jogar a carta no campo de batalha. Um exemplo do baralho e do campo de combate pode ser visto na Figura 9.

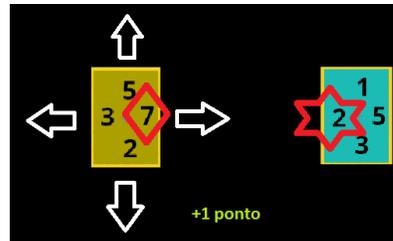
Figura 9 - Baralhos e Campo de combate.



Fonte: *Elaborado pelo autor: Giulianno, 2023.*

O poder do ataque de carta de monstro depende de onde ela foi jogada no campo de combate. Quando uma carta é jogada perto da carta do oponente, ela irá começar um ataque na direção da carta do oponente e caso o poder de ataque daquela direção for mais forte que a carta do adversário que já existe no campo de combate, a carta derrotada será destruída e jogada para o campo de cemitério, com isso o atacante receberá mais 1 ponto. O jogo termina quando nenhum jogador tiver mais cartas para jogar ou se não existir mais espaços no campo de combate para aceitar as cartas. Vence o jogo aquele quem tiver o maior número de pontos. Um exemplo do ataque e ganho de pontos pode ser visto na Figura 10.

Figura 10 - Ataque e ganho de pontos.



Fonte: *Elaborado pelo autor: Giulianno, 2023.*

3.2 MOTOR GRÁFICO

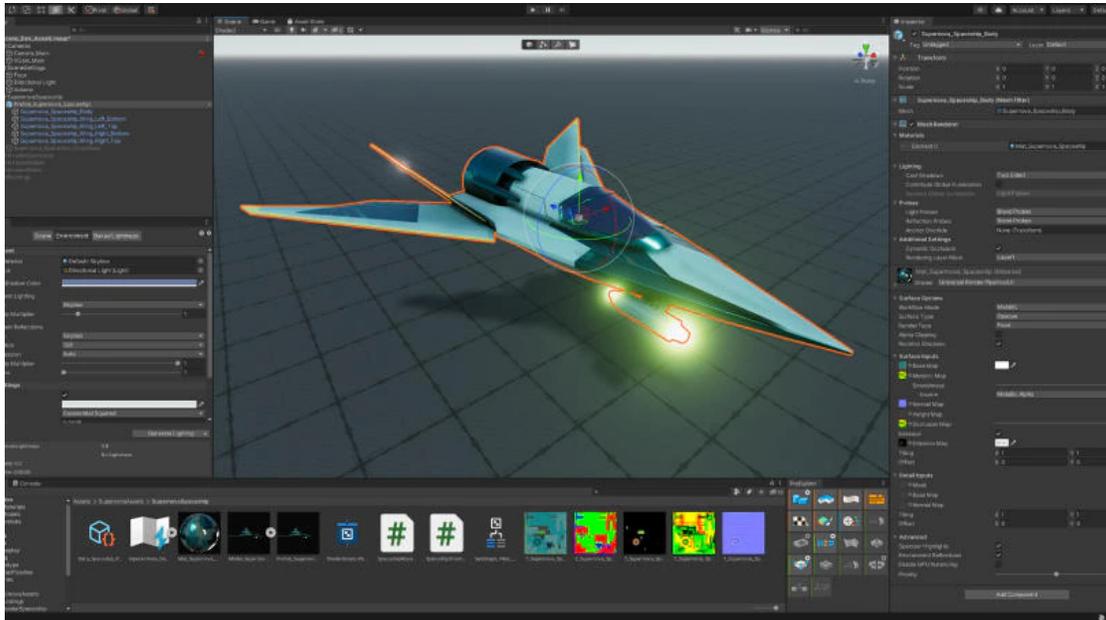
Durante a criação da ideia do jogo *CardBattle*, foi pensado qual seria a melhor linguagem, *framework* ou motor gráfico a ser utilizado. No mercado, existem diversos tipos de motores gráficos disponíveis, que podem ser de grande ajuda para os pequenos desenvolvedores em suas criações.

3.2.1 UNITY

O *Unity* é um motor gráfico 2D e 3D que foi criado pela Unity Technologies em 8 de junho de 2005 tornando-se gratuito em 2009, mas ainda possuindo alguns planos pagos e é considerado o motor gráfico mais utilizado no mundo para o desenvolvimento de jogos, mas o seu uso é um pouco mais complicado e exige que o usuário tenha um bom conhecimento da

linguagem de programação C Sharp. Um exemplo de como é a interface do *Unity* pode ser visto na Figura 11.

Figura 11 - Exemplo da interface de produção do motor gráfico do *Unity*.



Fonte: Unity, 2022.

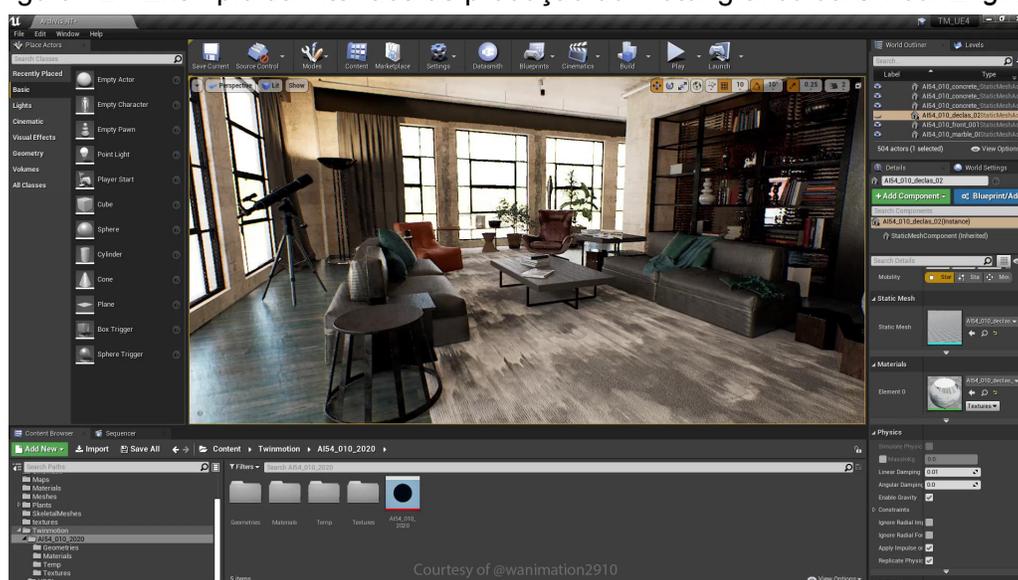
O motor gráfico *Unity* é um sistema multi-plataforma e possui suporte para os seguintes sistemas:

- *BlackBerry 10*
- *Windows*
- *Microsoft HoloLens*
- *OS X*
- *Linux*
- *Android*
- *IOS*
- *WebGL*
- *Oculus Rift*
- *ARCore*
- *Stadia*
- *Magic Leap*
- *PlayStation Vita, 4 e 5*
- *Xbox 360, One e Series X|S*
- *Nintendo Wii, Wii U, 3DS e Switch*

3.2.2 UNREAL ENGINE

A *Unreal Engine*, é um motor gráfico 3D que foi criado em 22 de maio de 1998 e é considerado o segundo motor gráfico 3D gratuito mais utilizado no mundo onde o foco maior deste motor gráfico está no desenvolvimento de jogos 3D mais realistas. A complexidade do uso desta ferramenta varia de acordo com o projeto, pois é possível desenvolver uma proposta mais complexa utilizando a linguagem de programação C++ ou criar um objeto sem precisar de um avançado conhecimento em programação, pois a *Unreal Engine* possui um sistema de Drag-and-Drop que facilita bastante o desenvolvimento, mas que provavelmente irá sacrificar o desempenho do produto final. Um exemplo de como é a interface da *Unreal Engine* pode ser visto na Figura 12.

Figura 12 - Exemplo da interface de produção do motor gráfico da *Unreal Engine*.



Fonte: *Unreal Engine*, 2004.

O motor gráfico *Unreal Engine* é um sistema multi-plataforma e possui suporte para os seguintes sistemas:

- *Windows*
- *PlayStation 4 e 5*
- *Xbox One, Series X|S*
- *Nintendo Switch*
- *Stadia*
- *macOS*
- *iOS*
- *Android*
- *ARKit*
- *ARCore*
- *OpenXR*
- *SteamVR*

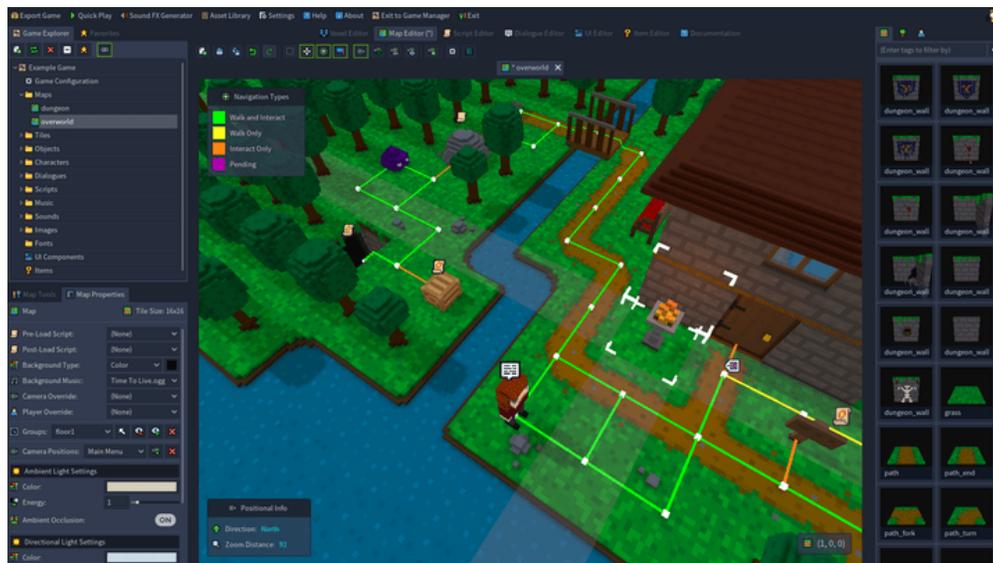
- *Oculus*
- *Linux*
- *SteamDeck.*

3.2.3 GODOT

O *Godot* é um o motor gráfico 2D e 3D gratuito de código aberto, desenvolvido por Juan Linietsky e Ariel Manzur junto com a OKAM Studio em 2001 e em 15 de dezembro de 2014 a primeira versão estável foi disponibilizada para o público poder utilizá-la. Esta ferramenta é utilizada por desenvolvedores menores, e que vem se destacando bastante no mercado de jogos indie pelo seu uso bem simples e fácil. Esta ferramenta trabalha com 4 diferentes linguagens de programação, sendo elas:

- GD Script: que é uma linguagem própria da plataforma, de fácil entendimento e que lembra bastante a linguagem de programação Python
- C Sharp que foi adicionado nas novas versões para atender os requisitos padronizados que o mercado de desenvolvimento de jogos requisitava
- Visual Script, que é uma programação feita diretamente pela interface gráfica através de Drag-and-Drop, pensado nas pessoas que não possuíam conhecimentos na área da programação.
- GD Native que é uma linguagem que lembra bastante o C++, desenvolvida para a modificação do próprio motor gráfico, possibilitando o desenvolvimento de extensões para a plataforma. Um exemplo de como é a interface do *Godot* pode ser visto na figura 13.

Figura 13 - Exemplo da interface de produção do motor gráfico do *Godot*.



Fonte: *Godot*, 2007.

O motor gráfico conhecido como *Godot* é um sistema multi-plataforma e possui suporte para os seguintes sistemas:

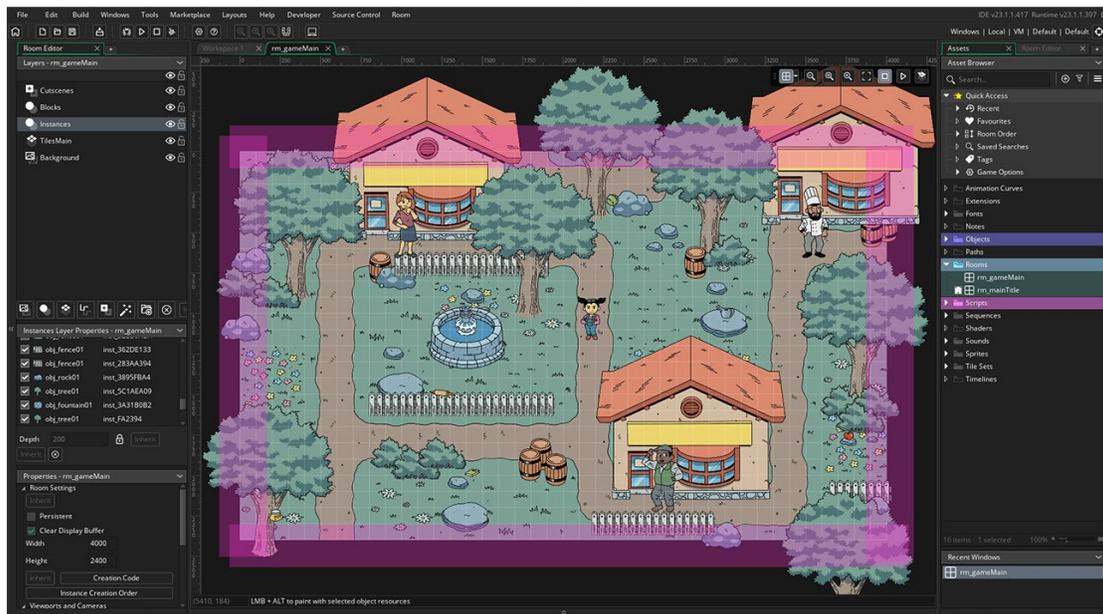
- *Windows*
- *macOS*
- *Linux*
- *UWP*
- *BSD*
- *IOS*
- *Android*
- *Nintendo Switch*
- *PlayStation 4*
- *Xbox One*
- *HTML 5*
- *WEB Assembly*

3.2.4 GAME MAKER STUDIO 2

Pelo fato deste projeto ser feito apenas por uma única pessoa, então se deu a vontade de recorrer a algo que pudesse ser produzido em um espaço de tempo mais curto e com uma ferramenta onde tivesse um maior conhecimento e afinidade, logo, o motor gráfico *Game Maker Studio 2* foi escolhido para ajudar neste papel, mesmo sendo um motor gráfico pago.

O *Game maker Studio 2* é um mecanismo de desenvolvimento de jogos criado por Mark Overmars em 15 de novembro de 1999, que possui a sua própria linguagem de scripts chamada de Gamemaker Language ou também conhecido como GML que lembra um pouco a linguagem de programação Python. Ao longo dos anos o Game Maker 2 vem se destacando um pouco no desenvolvimento de jogos independentes e que possuem uma equipe de trabalho bem pequena. É possível criar diferentes tipos de jogos com este mecanismo de desenvolvimento, limitando o desenvolvedor apenas com a sua própria criatividade. Apesar do *software* ser um motor de criação de jogos 2D é possível criar elementos gráficos em 3D, mas será baseado em efeitos visuais rígidos e não em um formato mais profundo e vivo. Mesmo assim ele é um excelente artifício para o desenvolvimento de projetos 2D. Um exemplo de como é a interface do programa pode ser visto na figura 14.

Figura 14 - Exemplo da interface de produção do motor gráfico do *Game Maker 2*.



Fonte: *GameMaker*, 2022.

O motor gráfico *Game Maker Studio 2* é um sistema multi-plataforma e possui suporte para os seguintes sistemas:

- *Steam*
- *Itch.io*
- *GX.games*
- *Windows*
- *Mac OS*
- *Ubuntu*
- *HTML5*
- *IOS*
- *Amazon Fire*
- *Android TV*
- *TvOS*
- *Nintendo Switch*
- *PlayStation 4 e 5*
- *Xbox One, Series X|S*

3.3 ESTRUTURA DE DADOS

Durante o desenvolvimento de um jogo de cartas customizado, a aplicação adequada da Estrutura de Dados desempenha um papel fundamental na eficiência e funcionalidade do jogo. De acordo com Thomas Cormen (2012), a Estrutura de Dados é responsável por organizar e gerenciar os elementos do jogo, permitindo a manipulação e o armazenamento eficiente de diferentes tipos de dados com tamanhos variados.

No contexto específico de um jogo de cartas, a Estrutura de Dados desempenha um papel crucial em várias etapas do jogo. Uma das aplicações é na representação do baralho, onde as cartas são armazenadas e organizadas. Uma abordagem comumente utilizada é a estrutura de pilha, em que as cartas são empilhadas umas sobre as outras. Isso permite que o baralho seja facilmente acessado, adicionado ou removido, replicando a mecânica de um baralho físico.

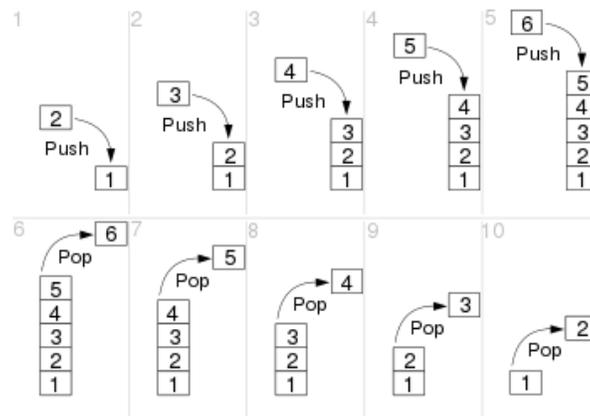
Além disso, ela é essencial para a manipulação das cartas durante o jogo. Por exemplo, no momento do embaralhamento do baralho, a Estrutura de Dados pode ser utilizada para randomizar a ordem das cartas, garantindo um elemento de aleatoriedade no jogo.

No campo de combate, técnicas podem ser aplicadas para armazenar as informações das cartas jogadas. Uma representação comum é o uso de uma grade ou matriz, em que cada posição representa uma célula na qual uma carta pode ser posicionada. Dessa forma, a Estrutura de Dados possibilita a identificação rápida da posição de cada carta no campo de combate e o acesso aos dados relevantes associados a ela, como atributos de ataque, defesa ou efeitos especiais.

Jogos de cartas customizados podem ter a presença de uma inteligência artificial para jogar contra o jogador. Nesse caso, a Estrutura de Dados também desempenha um papel importante. Por exemplo, a máquina pode escanear a matriz que representa o campo de combate, analisando as posições ocupadas pelas cartas e identificando as jogadas que pode fazer com base nas regras do jogo. Essa análise da Estrutura de Dados permite que a máquina tome decisões estratégicas e ofereça um desafio adequado ao jogador, tornando a experiência mais envolvente e competitiva.

Cada estrutura oferece benefícios específicos e pode ser aplicada de acordo com as características e regras do jogo de cartas customizado. No desenvolvimento do jogo, é importante considerar a eficiência da manipulação de dados. A escolha adequada da Estrutura de Dados pode contribuir para uma melhor performance e escalabilidade do jogo, permitindo um processamento rápido e eficiente das ações dos jogadores, como jogar cartas, calcular resultados de combate ou verificar condições de vitória. Um exemplo de como a estrutura de dados pode ser usada pode ser visto nas figuras 15 e 16.

Figura 15 - Empilhando e desempilhando dados.



Fonte: Wikipedia, 2022

Figura 16 - Ilustração da definição de um elemento em uma grade.

		ds_grid x axis				
		0	1	2	3	4
ds_grid y axis	0	4	5	3	1	6
	1	5	4	3	7	7
	2	0	3	4	3	5
	3	4	9	8	3	2
	4	2	2	2	1	0

Fonte: *GameMaker*, 2022.

Um jogo de baralho lida bastante com a sorte, então é muito importante garantir que a aleatoriedade dos valores sempre aconteça, assim, o baralho do jogador constantemente entregará cartas de forma incerta e podemos certificar esta funcionalidade com uma técnica conhecida como aleatoriedade de sementes, que faz com que os valores gerados tentem não se repetir, Cormen (2012).

Em resumo, a aplicação correta da Estrutura de Dados em um jogo de cartas customizado desempenha um papel vital na organização, manipulação e gerenciamento eficiente dos dados do jogo. Através de estruturas como pilhas, matrizes ou outras estruturas mais complexas, é possível criar uma base sólida para o desenvolvimento do jogo, garantindo uma experiência de jogo fluida, funcional e imersiva para os jogadores.

3.4 TESTES

Logo após a criação do conceito do jogo *CardBattle*, foi decidido como os testes poderiam ser realizados até a concretização do produto com a ajuda de uma técnica de discussão conhecida como *BrainStorm* que foi feita junta de um grupo de amigos. A avaliação do objeto será realizada em um computador com as seguintes configurações: *core i7 2600*, 16g ram, *GeForce GTX 1050*, *GeForce GTX 1650*, *Windows 10 pro 21h2*, utilizando o motor gráfico *GameMaker Studio 2* com uma licença gratuita, mas que futuramente necessitará do uso de uma licença paga para pc, assim, podendo gerar um executável .exe que facilitará o teste do jogo *CardBattle* em computadores de amigos próximos, verificando a possibilidade de possíveis problemas que possam existir, como por exemplo: Defeitos causados pela diferença nas versões de sistemas operacionais, tamanhos de monitor e entre outros.

4. DESENVOLVIMENTO DO PROJETO

Neste capítulo, será discutido o processo de desenvolvimento do protótipo de baixa fidelidade do jogo de cartas com um jogador humano e uma inteligência artificial, bem como a transição deste protótipo para o *GameMaker 2*. A fim de criar uma versão funcional do jogo, foi necessário seguir uma série de etapas que permitiram identificar os elementos essenciais do jogo e testar diferentes funcionalidades.

4.1 DESENVOLVIMENTO DO PROJETO DE BAIXA FIDELIDADE

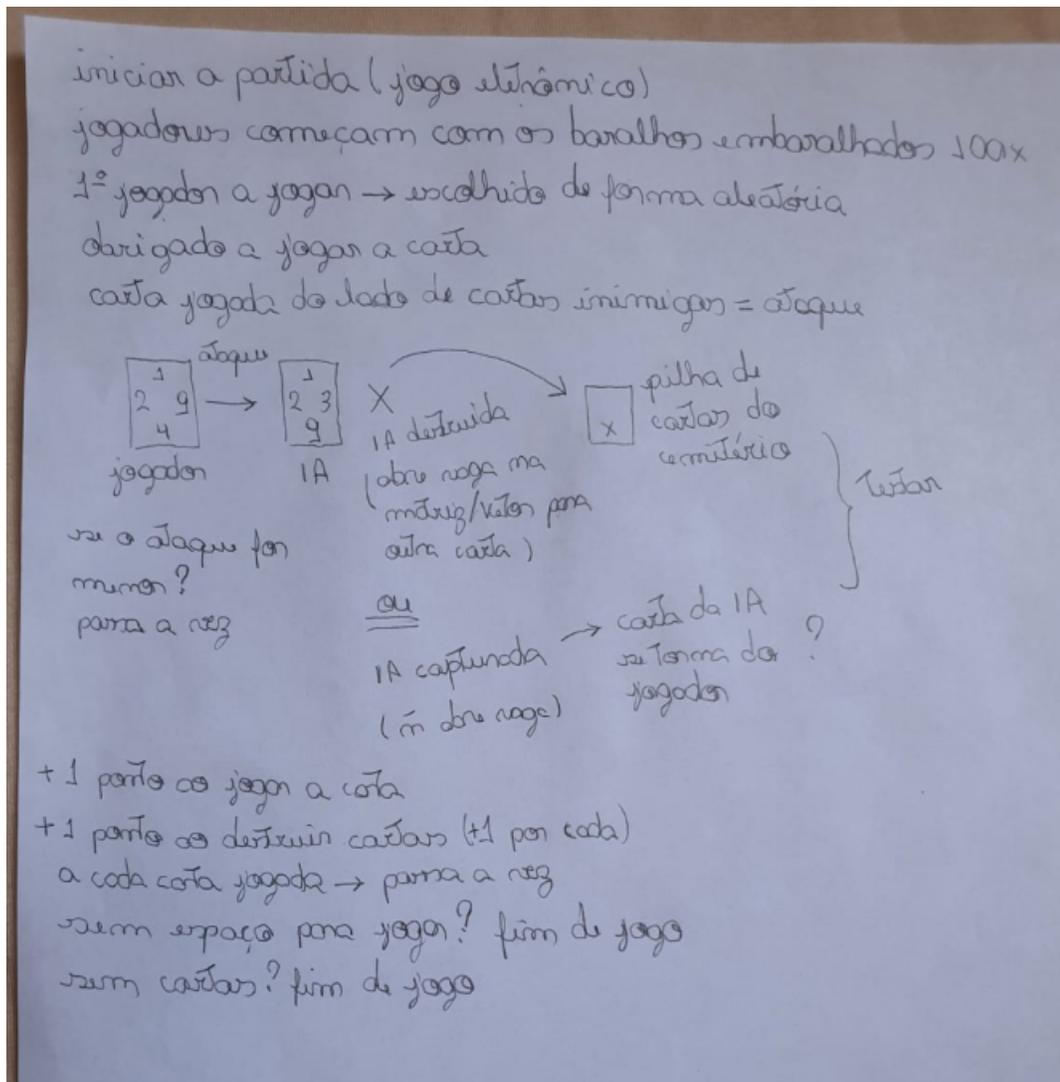
O primeiro passo para o desenvolvimento do protótipo de baixa fidelidade foi criar uma lista com as principais funcionalidades que o jogo deveria ter. Então foi realizado um procedimento conhecido como *Brainstorm*, a técnica foi realizada com um grupo de amigos onde a ideia é colocar no papel tudo aquilo que passar pela cabeça dos desenvolvedores sobre o projeto e o que poderia ser interessante de utilizar.

Geralmente durante o *Brainstorm*, não se deve pensar muito se aquilo é viável ou não, apenas colocar no papel tudo o que imaginar que poderia existir no produto (ALBERTO, 2015). Como por exemplo: Qual o nome do jogo? O jogo será exatamente sobre o que? Ele terá inteligência artificial? Quantos jogadores podem jogar o jogo simultaneamente? Terá níveis de dificuldade? Terá algum modo online? Tudo o que passar pela cabeça do desenvolvedor deve ser colocado no papel e então depois disso, cada coisa colocada será analisada para decidir se realmente aquilo será útil ou não no produto final. O *brainstorm* pode ser visto na figura 17.

- Qual motor gráfico usar? *Game maker 2*
- Qual o nome do jogo? *CardBattle*
- O jogo possui alguma história?
- O jogo possui um público alvo?
- Idade mínima para jogar?
- Em quais plataformas o jogo irá funcionar? PC
- Como o jogo funciona?
- Quais os objetivos?
- Regras do jogo?
- Terá modo *Online*? Não
- Marcação de pontos? Sim
- Quantos jogadores? Dois
- Terá Inteligência Artificial? Sim
- A Inteligência Artificial terá dificuldades?
- Quais Dificuldades? Normal
- Quem são os jogadores? Humano vs Máquina
- Como será o campo de combate? 4x4, slot para baralho, slot para cemitério
- Quais os tipos de cartas? Lendária, Monstros de ataque
- Quantas cartas cada baralho deve ter? 30 (29+1)
- Quais as regras do baralho? 1 Lendária, 29 Monstros
- Estrutura de dados? Pilha, Fila, *Shuffle* (Embaralhar), *Seed*, *Árvore*, *Grid/Matriz/Vetor*
- Pilha: As cartas serão empilhadas para formar o baralho
- Fila: A carta usada será sempre a do topo do baralho
- *Shuffle*: Embaralhar o baralho no início da partida
- *Seed*: As cartas do baralho devem ser criados de forma aleatória
- *Árvore*: A forma como a IA tomará decisões
- *Grid/Matriz/Vetor*: O campo de combate

Foi também desenvolvido outra lista, que incluía itens como a escolha aleatória do jogador que começaria a jogar, ao iniciar a partidas cada jogador teria o seu baralho embaralhado de forma aleatória, a colocação de cartas no campo de batalha, a seleção de ataques em diferentes direções, a pontuação, a captura ou destruição de cartas. A lista de ideias pode ser vista na figura 18.

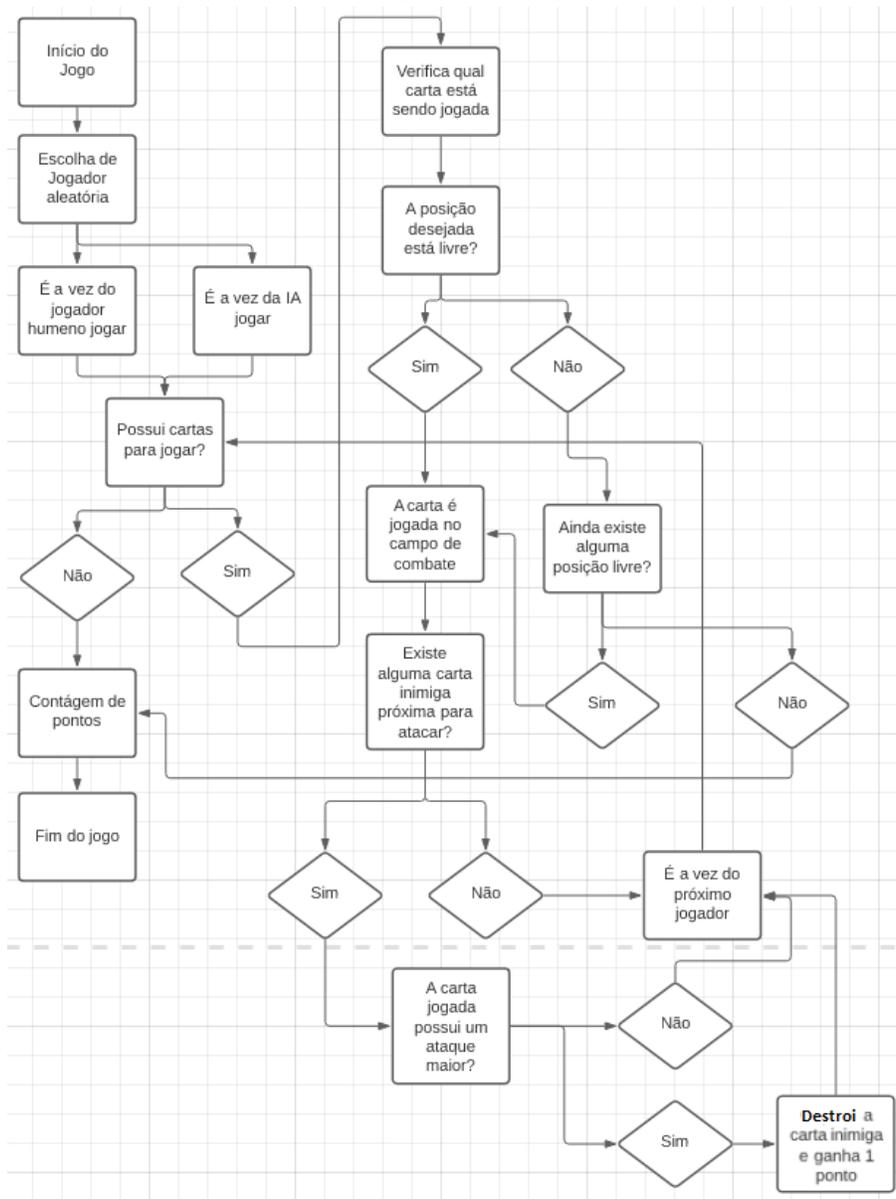
Figura 18 - Ideia de regras e funcionalidades.



Fonte: Elaborado pelo autor: Giulianno, 2023.

Em seguida, foi desenvolvido um fluxograma usando o *software* de criação de diagramas chamado *Draw.io*. O fluxograma inclui uma representação um pouco detalhada de como o jogo funcionaria. Esse protótipo permitiu testar diferentes ideias. O fluxograma pode ser visto na figura 19.

Figura 19 - Fluxograma do jogo *Card Battle*.



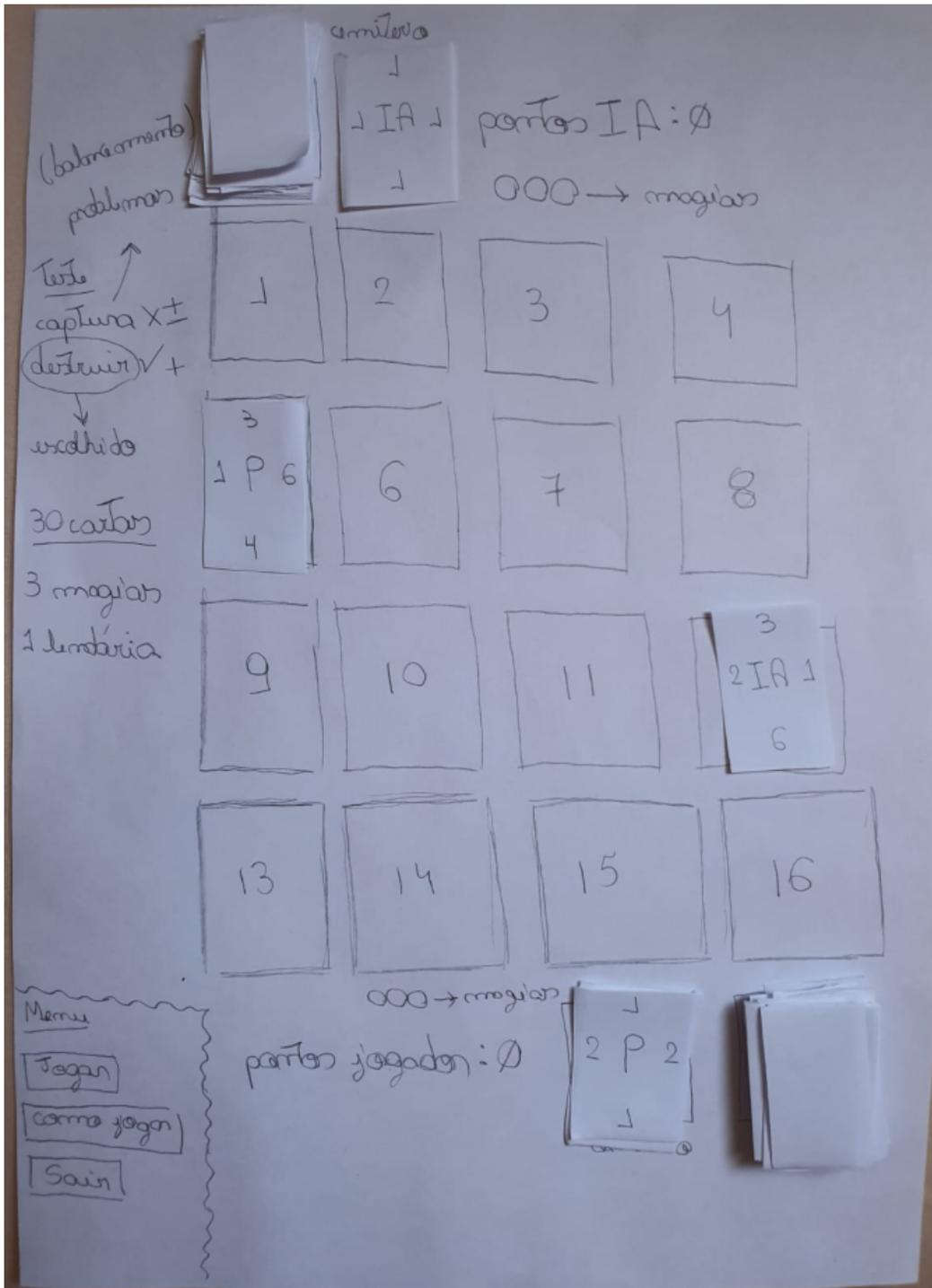
Fonte: Elaborado pelo autor: *Giuliano*, 2023.

1. O jogo inicia.
2. O jogo escolhe aleatoriamente entre o jogador humano ou a máquina, quem irá iniciar jogando.
3. O jogador escolhido é obrigado a jogar uma carta do topo do seu baralho.
4. Como é a primeira jogada, todas as posições no campo de combate estão válidas para se jogar uma carta.
5. A vez é passada para o outro jogador.
6. O próximo jogador é obrigado a jogar a carta do topo do seu baralho.
7. Se o jogador tentar jogar a carta em uma posição inválida, ele deve verificar se ainda existem posições válidas para jogar.

8. Se houver posições válidas, o jogador joga a carta em uma posição válida.
9. Se o jogador jogar a carta ao lado da carta do inimigo, inicia-se o ataque.
10. Se o valor do ataque for maior do que o valor da defesa da carta inimiga, a carta inimiga é destruída e a carta destruída é enviada para o cemitério, o jogador ganha mais um ponto por cada carta destruída e passa a vez.
11. Se o valor do ataque não for maior que o valor da defesa da carta inimiga, o jogador apenas passa a vez.
12. O ciclo de jogadas se repete até que nenhum jogador tenha mais cartas em seu baralho para jogar ou não existam mais posições válidas para jogar.
13. O jogo termina e é realizada a contagem dos pontos para determinar o vencedor.

Também foi criado um esboço rápido do jogo em papel, que consistia em desenhos simples de cartas e do campo de batalha, com anotações sobre as diferentes funcionalidades do jogo. Esse esboço foi utilizado para testar diferentes cenários, como a colocação de cartas em diferentes posições do campo de batalha e a resolução de ataques. O esboço pode ser visto na Figura 20.

Figura 20 - Esboço do teste do jogo CardBattle.



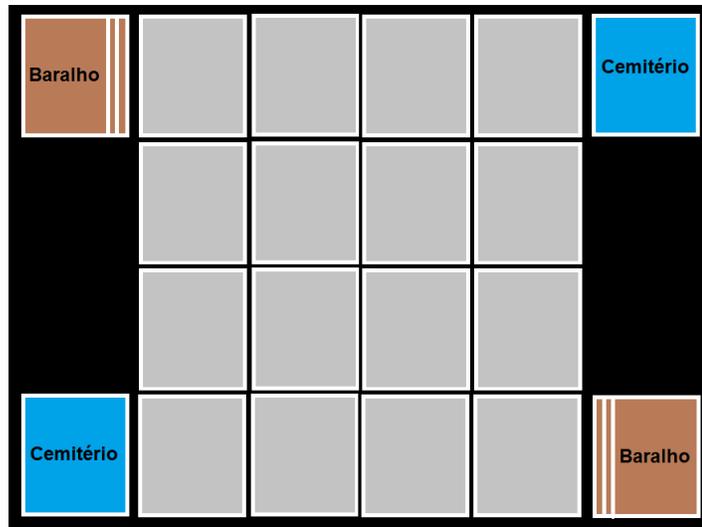
Fonte: Elaborado pelo autor: Giuliano, 2023.

4.2 O PROCESSO DE DESIGNER

De acordo com Scott Rogers (2014), toda a criação de um jogo se inicia pelo *designer* e deve-se fazer simples perguntas para conseguir obter o desejado jogo. O que é o jogo que está sendo criado? Será para quantos jogadores? Quais as regras que existem neste jogo? Qual é a condição de vitória? Qual é o objetivo? Não esquecendo de sempre manter as suas respostas o mais simples possível.

O desenvolvimento para os primeiros protótipos do jogo *CardBattle* foram elaborados logo após a decisão do que seria ou não usado depois do *BrainStorm* e do esboço rápido do cenário do jogo, logo após isso, foi criado um protótipo melhor elaborado para começar o pensamento de como seria todo o *designer* pronto do jogo. O jogo deveria ser algo simples e rápido de ser entendido em poucas jogadas, mas também que conseguisse beirar a complexidade para os jogadores mais estratégicos, dando a oportunidade de criar estratégias de acordo com as cartas que recebia. O primeiro protótipo do campo de combate foi pensado em um tamanho de 4x4 onde cada jogador deve escolher em qual posição ele desejaria manter a sua carta. O campo de combate pode ser visto na Figura 21.

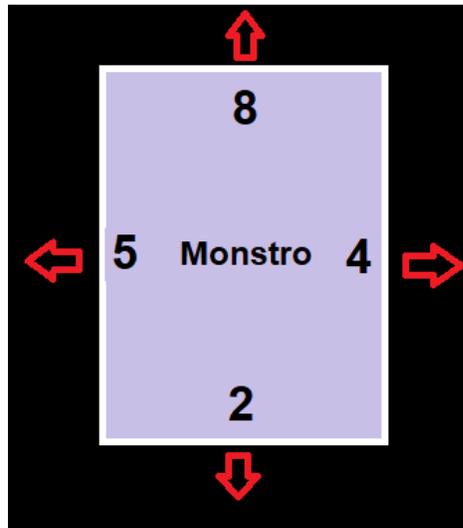
Figura 21 - Ilustração do campo de combate do jogo *CardBattle*.



Fonte: Elaborado pelo autor: *Giuliano*, 2022.

No início de cada partida o baralho de cada jogador deve ser embaralhado para que cada um recebesse cartas aleatórias, após o embaralhamento, cada jogador irá obter uma carta do topo de seu baralho e ele é obrigado a jogar esta carta no campo de combate. No campo de combate serão posicionadas as chamadas de Cartas de Monstros, que são as cartas utilizadas para realizar os ataques contra as cartas do oponente. Cada carta de monstro possui os seus próprios ataques, onde cada ataque depende de como a carta é posicionada no campo de combate, determinando a direção do ataque. Um exemplo da carta de monstro pode ser visto na Figura 22.

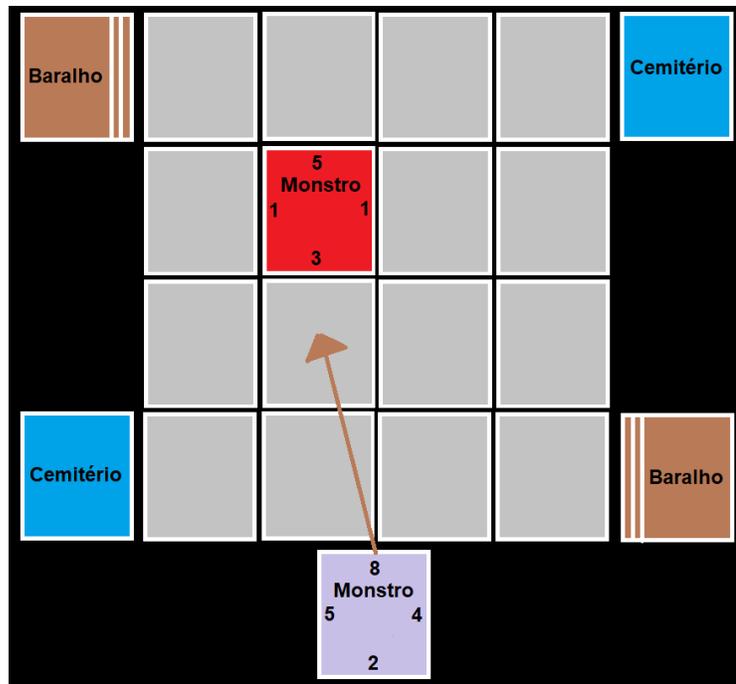
Figura 22 - Ilustração da carta de Monstro.



Fonte: Elaborado pelo autor: Giulianno, 2022.

O oponente posicionou uma carta de monstro no campo de combate e agora é a vez do jogador jogar a sua carta. O jogador irá atacar a carta do adversário posicionando ela de uma forma que o seu ataque seja maior que o lado que será atacado. É possível ver um exemplo na Figura 23.

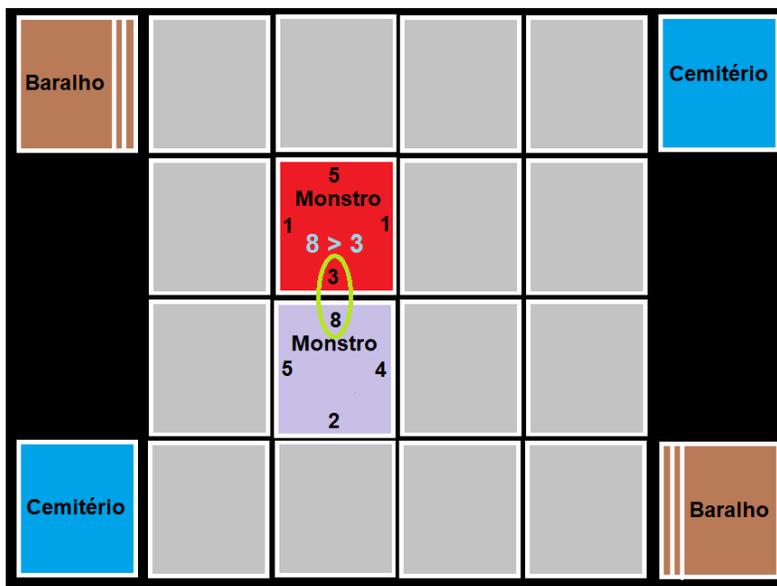
Figura 23 - Ilustração da carta de Monstro atacando.



Fonte: Elaborado pelo autor: Giulianno, 2022.

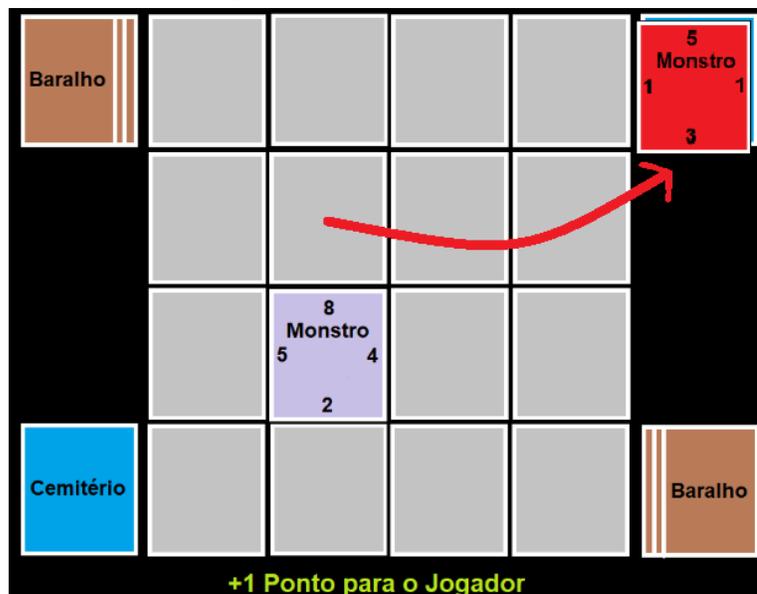
Como o número de ataque 8 é maior que o número de ataque 3, a carta do jogador vence, fazendo com que ele receba +1 ponto e então a carta do adversário é destruída e enviada para o cemitério e então será a vez do outro jogador de realizar a sua jogada. Caso não for mais possível jogar nenhuma carta no campo de combate, seja por falta de cartas no baralho ou falta de espaço no campo de combate, o jogo termina e vence o jogador que tiver o maior número de pontos. É possível ver um exemplo nas Figuras 24 e 25.

Figura 24 - Ilustração da carta de Monstro atacando.



Fonte: Elaborado pelo autor: Giulianno, 2022.

Figura 25 - Ilustração da carta de Monstro atacando.



Fonte: Elaborado pelo autor: Giulianno, 2022.

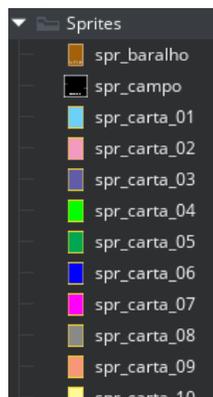
4.3 TRANSIÇÃO PARA O GAMEMAKER 2

Depois de testar o protótipo de baixa fidelidade e confirmar que as principais funcionalidades estavam funcionando corretamente, foi iniciada a transição para o *GameMaker 2*. Ele foi escolhido porque ele permite que jogos sejam criados rapidamente e tem uma curva de aprendizado relativamente suave. O processo de transição envolveu a criação de *sprites*, objetos e *scripts* para replicar as funcionalidades testadas no protótipo.

Primeiramente, foi trabalhada na interface do usuário do jogo. Foi usado um editor do *GameMaker 2* para criar as diferentes telas do jogo, incluindo a tela principal, a tela de jogo e a tela de vitória. Em seguida, foi adicionado diferentes objetos ao jogo, como cartas, salas que irão aparecer no jogo, seleção de menu e outras imagens.

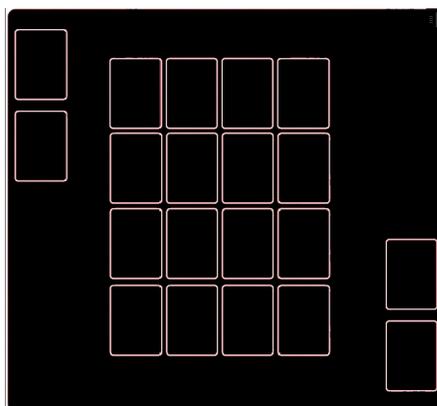
Esta etapa da transição também envolveu a criação dos *sprites*, que representavam as cartas e o campo de batalha. Foram criados *sprites* para cada tipo de carta, bem como para o campo de batalha, que foram criados em um *software* de edição de imagens e, em seguida, importados para o *GameMaker 2*. É possível ver um exemplo nas Figuras 26 e 27.

Figura 26 - *Sprites* das cartas.



Fonte: Elaborado pelo autor: *Giuliano*, 2022.

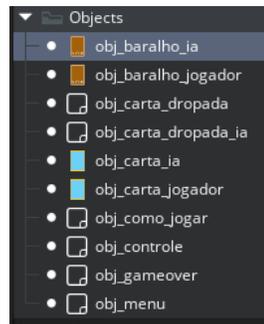
Figura 27 - *Sprite* do campo de batalha.



Fonte: Elaborado pelo autor: *Giuliano*, 2022.

Logo após isso foram criados objetos para cada um desses *sprites*, que foram usados para implementar as diferentes funcionalidades do jogo. Os objetos incluíam o baralho, a carta, o campo de batalha e o jogador humano e inteligência artificial. Cada objeto tinha diferentes *scripts* que permitiam o embaralhamento dos baralhos, a colocação de cartas no campo de batalha, a seleção de ataques e a contagem de pontos. É possível ver um exemplo na Figura 28.

Figura 28 - Objetos utilizados.



Fonte: Elaborado pelo autor: *Giuliano*, 2022.

Depois dos *sprites* estarem prontos, o próximo passo era realizar a elaboração de diversos *scripts* com o objetivo de garantir o correto funcionamento das funcionalidades. Esses *scripts* desempenharam diferentes papéis para permitir a interação adequada dos jogadores com o jogo.

Um dos *scripts* criados possibilitou que o jogador pudesse escolher qual carta jogar e em qual posição do campo de batalha ela seria colocada. Essa funcionalidade era essencial para que o jogador tivesse controle sobre suas jogadas e pudesse tomar decisões estratégicas durante o jogo.

Além disso, um outro *script* foi desenvolvido para permitir a interação direta entre as cartas. Esse script possibilitou que as cartas interagissem entre si de acordo com suas características e regras estabelecidas no jogo. Essa interação entre as cartas proporcionou uma dinâmica mais complexa e desafiadora para os jogadores.

Após a implementação dessas funcionalidades principais, foi iniciado o trabalho na inteligência artificial do jogo. Utilizando um *script* específico, foi possível desenvolver uma inteligência artificial capaz de escolher qual carta jogar e em qual posição do campo de batalha posicioná-la. Além disso, a inteligência artificial foi programada para interagir corretamente com as cartas controladas pelo jogador humano. Essa etapa permitiu a criação de partidas contra o computador, oferecendo aos jogadores a possibilidade de enfrentar um oponente virtual.

Por fim, a lógica do jogo foi trabalhada através da criação de um *script* responsável por determinar as condições de encerramento da partida e identificar o jogador vencedor. Esse *script* estabeleceu critérios e regras para determinar quando o jogo terminaria, levando em consideração fatores como pontos e outras condições específicas definidas para o jogo em questão.

Dessa forma, em resumo, por meio da elaboração e implementação de diferentes *scripts*, foi possível desenvolver o protótipo funcional do jogo, com suas funcionalidades

essenciais, inteligência artificial, interação entre as cartas e a lógica para determinar o término e o vencedor da partida.

5. RESULTADOS

No quinto capítulo deste trabalho, será discutido a mecânica e o funcionamento do jogo *CardBattle* em seu estágio de construção final. Como um jogo de cartas estratégico, o *CardBattle* oferece uma experiência envolvente, desafiadora e divertida para os jogadores. Neste capítulo, exploraremos as principais características do jogo, incluindo a tela inicial, os jogadores, o baralho de cartas, o campo de combate e as regras do jogo.

5.1 Tela Inicial e Menu

Ao iniciar o *CardBattle*, os jogadores são recebidos por uma tela inicial com um menu simples de se utilizar. O menu contém três opções principais: "Iniciar Jogo", "Como Jogar" e "Sair". Os jogadores podem selecionar "Iniciar Jogo" para começar uma nova partida, "Como Jogar" para obter instruções sobre as regras do jogo ou "Sair" para encerrar o aplicativo. É possível ver um exemplo na Figura 29.

Figura 29 - Menu.



Fonte: Elaborado pelo autor: Giulianno, 2022.

5.2 Jogadores

O *CardBattle* é um jogo para dois jogadores, sendo um jogador humano e um jogador controlado por uma inteligência artificial (IA). A interação entre esses dois jogadores é o objetivo central do jogo, oferecendo desafios estratégicos e táticos ao longo da partida.

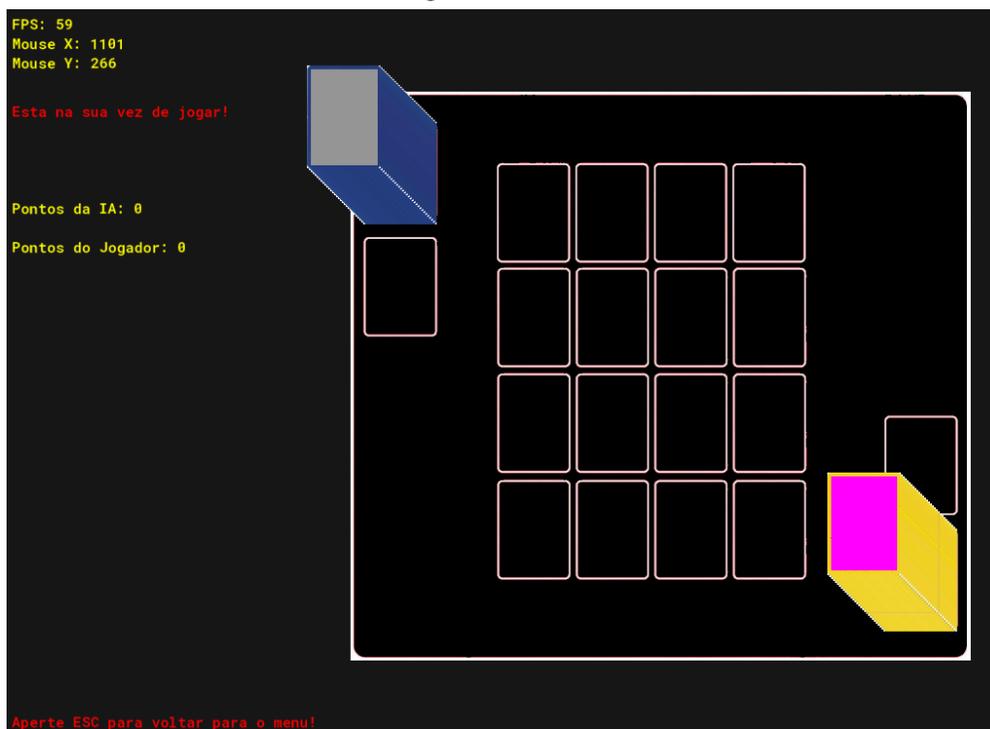
A inteligência artificial foi programada para tomar decisões com base em uma lógica pré-definida, simulando um desafio apropriado para um jogador mediano. Isso proporciona um

nível adequado de dificuldade e torna a experiência de jogo interessante tanto para jogadores casuais quanto para aqueles que procuram desafios mais complexos.

5.3 Baralho de Cartas

Cada jogador possui um baralho de cartas exclusivo, composto por 30 cartas (29 cartas comuns e 1 carta lendária com atributos mais fortes). Antes do início do jogo, as cartas de cada baralho são embaralhadas automaticamente, garantindo uma distribuição aleatória das cartas durante a partida. Essa aleatoriedade contribui para a imprevisibilidade e a estratégia do jogo. É possível ver um exemplo na Figura 30.

Figura 30 - Baralho.



Fonte: *Elaborado pelo autor: Giulianno, 2022.*

5.4 Campo de Combate

O campo de combate é onde as cartas dos jogadores são colocadas em batalha. Cada jogador pode jogar uma carta do topo do seu baralho em uma posição livre do campo de combate onde as cartas do jogador recebem uma cor mais esverdeada enquanto as cartas da inteligência artificial recebem uma cor mais avermelhada. O objetivo é posicionar as cartas de forma estratégica para obter vantagem sobre o oponente. É possível ver um exemplo na figura 29.

5.5 Pontuação e Ataques

Cada carta do baralho possui diferentes tipos de ataques, que podem ser direcionados para cima, para baixo, para a esquerda ou para a direita. Durante o jogo, os jogadores podem atacar as cartas do oponente já presentes no campo de combate. Por exemplo, se o jogador deseja atacar a parte de baixo da carta do oponente, ele utilizará o valor de ataque superior de sua própria carta para confrontar o valor de ataque inferior da carta do oponente.

Cada carta colocada com sucesso no campo de combate garante 1 ponto para o jogador da vez e caso o número de ataque do jogador for maior do que o valor de ataque do oponente, o jogador ganha um ponto adicional e a carta do oponente é destruída, sendo descartada no monte chamado de "cemitério". Essa mecânica de combate adiciona camadas estratégicas ao jogo, pois os jogadores devem escolher sabiamente quando e onde atacar, levando em consideração as características das cartas e as possibilidades de pontuação. É possível ver um exemplo na figura 29.

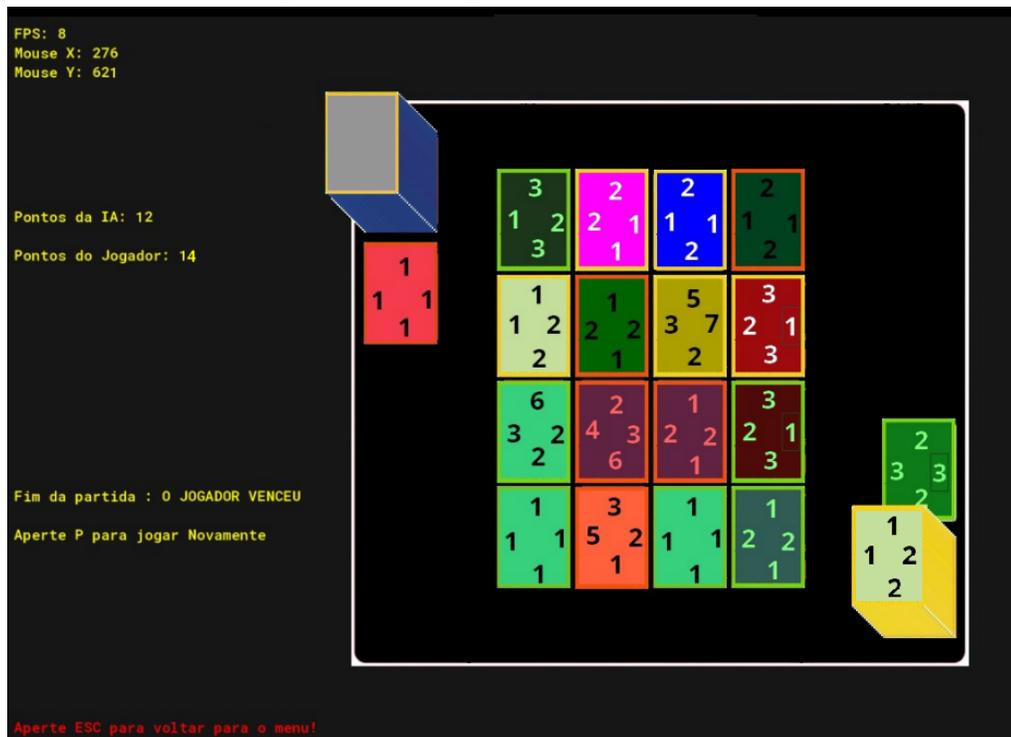
5.6 Condições de Vitória

O jogo *CardBattle* termina quando algum jogador não tiver mais cartas para jogar ou quando o campo de combate não tiver mais espaço disponível para receber cartas. Essas condições determinam o fim da partida e são cruciais para determinar o vencedor.

Se um jogador não tiver mais cartas para jogar, significa que esgotou todo o seu baralho. Nesse caso, a partida termina e então é feita a contagem dos pontos. É importante que os jogadores gerenciem suas cartas com cuidado, planejando suas jogadas para evitar ficar sem opções no decorrer da partida.

Além disso, se o campo de combate não tiver mais espaço para receber cartas, impede que ambos os jogadores coloquem novas cartas em jogo. Essa situação também indica o fim da partida, onde então é realizada a contagem dos pontos. O limite de espaço no campo de combate adiciona uma camada estratégica ao jogo, incentivando os jogadores a considerarem cuidadosamente suas ações e o posicionamento de suas cartas. É possível ver um exemplo na Figura 31.

Figura 31 - Campo de Combate.



Fonte: Elaborado pelo autor: Giulianno, 2022.

Em resumo, foi apresentado a mecânica e o funcionamento do jogo *CardBattle*. Através de sua tela inicial com menu, jogadores humanos e IA, baralhos de cartas, campo de combate e regras de pontuação, o jogo oferece uma experiência estratégica e competitiva.

Ao jogar *CardBattle*, os jogadores são desafiados a tomar decisões estratégicas ao posicionar suas cartas e escolher os ataques adequados para superar seu oponente. Com uma mecânica simples de aprender e uma variedade de possibilidades táticas, o jogo promete entretenimento e diversão aos seus jogadores.

6. CONSIDERAÇÕES FINAIS

Neste trabalho, foi apresentado o desenvolvimento de um jogo de cartas personalizadas, no qual um jogador humano enfrenta uma máquina, onde este primeiro pensamento foi de interesse em melhorar o conhecimento sobre o desenvolvimento de jogos eletrônicos e demonstrar o conhecimento obtido durante o curso de Ciência da Computação. O jogo foi criado utilizando o motor gráfico *GameMaker 2* e passou por testes em um computador com especificações mais do que adequadas, incluindo um processador *Core i7*, placas de vídeo *GTX 1050* e *GTX 1650*, e *16 GB* de RAM.

Durante o processo de desenvolvimento, diversos desafios foram enfrentados, principalmente na criação dos gráficos e na lógica do jogo, como por exemplo, a tentativa de otimizar da melhor forma possível a utilização dos vetores e também fazer com que diferentes pilhas e a matriz interagissem entre si. No entanto, esses obstáculos foram superados com

sucesso por meio da utilização de recursos online, da consulta a materiais de referência e do conhecimento conquistado durante o curso.

Foram realizados diversos testes, como por exemplo, forçar jogadas que não poderiam acontecer, se cada posição dos vetores estavam funcionando como deveriam, nos quais foram identificados alguns bugs que foram corrigidos, além de melhorias na jogabilidade terem sido implementadas. Todos os resultados desses testes foram satisfatórios, pois o jogo foi capaz de rodar sem problemas no computador de teste e proporcionou uma experiência agradável ao jogador. Foi pensado um sistema de balanceamento onde a grande maioria das cartas tinha ataques fracos e com uma boa variação destes números para cada carta e dando atributos altos apenas para as cartas lendárias.

Considerando os resultados obtidos, é possível afirmar que o jogo de cartas desenvolvido é funcional e divertido. Entretanto, há pontos que podem ser aprimorados para tornar o jogo ainda mais atrativo. Entre as possíveis melhorias, destaca-se a inclusão de mais tipos de cartas, o aperfeiçoamento da inteligência artificial e uma interface mais bonita e chamativa.

Para o futuro, os planos incluem a implementação de um modo de um segundo jogador humano com um possível modo online, permitindo que dois jogadores humanos possam competir entre si e também um novo sistema de cartas chamado de cartas mágicas, onde cada jogador poderá escolher uma carta que terá diferentes propriedades para usar contra o adversário, como por exemplo, destruir uma carta aleatória do adversário ou até mesmo obrigá-lo a jogar o topo direto no cemitério. Outra ideia proposta seria transformar a máquina em uma real inteligência artificial que iria trabalhar como uma árvore de decisão. Além disso, por ser um jogo de cartas personalizadas, também seria possível criar cartas físicas para colecionar e jogar, proporcionando uma experiência híbrida entre o mundo digital e o físico como geralmente acontece com a maioria dos jogos de cartas personalizadas.

Em resumo, o desenvolvimento desse jogo de cartas foi um processo desafiador e recompensador, onde eu aprendi diversas técnicas que não tinha conhecimento como também pude colocar em prática os meus próprios conhecimentos sobre desenvolver jogos. Eu fiquei muito feliz com o resultado final, que foi um jogo funcional e divertido, que poderá ser aprimorado ainda mais no futuro. A utilização do *GameMaker 2* possibilitou a criação eficiente e facilitada do jogo, além de ter proporcionado um valioso aprendizado no desenvolvimento de jogos em geral. Com os planos futuros, é visado ampliar a experiência dos jogadores e continuar explorando novas possibilidades nesse campo emocionante e em constante evolução dos jogos eletrônicos.

REFERÊNCIAS BIBLIOGRÁFICAS

ALBERTO, Carlos. Definindo Escopo em Projetos de Software. Novatec Editora (8 maio 2015).

CORMEN, Thomas. ALGORITMOS - TEORIA E PRÁTICA. GEN LTC; 3ª edição (10 abril 2012).

Dixit, Ludopedia. Disponível em: <https://www.ludopedia.com.br/jogo/dixit>. Acesso em: 9 de set. de 2022.

GAMEMAKER, Engine: Disponível em: <https://gamemaker.io/pt-BR> . Acesso em: 5 de set. de 2022.

Gil, Antonio Carlos, Como elaborar projetos de pesquisa. - 6. ed. - São Paulo: Atlas, 2017.

GODOT, Engine. Disponível em: <https://godotengine.org/> . Acesso em: 26 de set. de 2022.

GURNEY, William. PLAYING CARDS: HISTORY OF THE PACK AND EXPLANATIONS OF ITS MANY SECRETS. Spring Books, 1931.

Lhama, Ludopedia. Disponível em: <https://ludopedia.com.br/jogo/lhama>. Acesso em: 9 de set. de 2022.

Magic the Gathering, Sobre o jogo. Disponível em: <https://magic.wizards.com/pt-BR/intro>. Acesso em: 3 de set. de 2022.

Magic, Ludopedia. Disponível em: <https://ludopedia.com.br/jogo/magic-the-gathering>. Acesso em: 3 de set. de 2022.

Magic the Gathering, Como jogar. Disponível em: <https://magic.wizards.com/pt-BR/how-to-play> . Acesso em: 3 de set. de 2022.

ROGERS, Scott. LEVEL UP: THE GUIDE TO GREAT VIDEO GAME DESIGNER. John Wiley & Sons; 2nd ed. edição (28 Abril 2014).

UNITY, Engine. Disponível em: <https://unity.com/pt> . Acesso em: 26 de set. de 2022.

UNO, Ludopedia. Disponível em: <https://ludopedia.com.br/jogo/uno>. Acesso em: 2 de set. de 2022.

UNREAL, Engine. Disponível em: <https://www.unrealengine.com/en-US> . Acesso em: 26 de set. de 2022.