

**UNIVERSIDADE FEDERAL DO MARANHÃO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
COORDENAÇÃO DE CIÊNCIA DA COMPUTAÇÃO
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

KLEYDSON BECKMAN BARBOSA

**DIRETRIZES DE GAME DESIGN E USABILIDADE PARA A CRIAÇÃO DE JOGOS
SÉRIOS USANDO TECNOLOGIAS WEB**

São Luís

2023

KLEYDSON BECKMAN BARBOSA

**DIRETRIZES DE GAME DESIGN E USABILIDADE PARA A CRIAÇÃO DE JOGOS
SÉRIOS USANDO TECNOLOGIAS WEB**

Monografia apresentada ao curso de Ciência da Computação da Universidade Federal do Maranhão, como parte dos requisitos necessários para obtenção do grau de Bacharel em Ciência da Computação.

Orientador: Prof. Dr. Carlos de Salles Soares Neto

São Luís
2023

Ficha gerada por meio do SIGAA/Biblioteca com dados fornecidos pelo(a) autor(a).
Diretoria Integrada de Bibliotecas/UFMA

Beckman Barbosa, Kleydson.

Diretrizes de Game Design e Usabilidade para a criação de jogos sérios usando tecnologias Web / Kleydson Beckman Barbosa. - 2023.

53 f.

Orientador(a): Carlos de Salles Soares Neto.

Monografia (Graduação) - Curso de Ciência da Computação, Universidade Federal do Maranhão, São Luis - MA, 2023.

1. Frameworks. 2. Game Design. 3. Jogos Sérios. 4. Tecnologias Web. 5. Usabilidade. I. de Salles Soares Neto, Carlos. II. Título.

DIRETRIZES DE GAME DESIGN E USABILIDADE PARA A CRIAÇÃO DE JOGOS SÉRIOS USANDO TECNOLOGIAS WEB

Monografia apresentada ao curso de Ciência da Computação da Universidade Federal do Maranhão, como parte dos requisitos necessários para obtenção do grau de Bacharel em Ciência da Computação.

Aprovado em: 16 / 11 / 2023.

BANCA EXAMINADORA

Prof. Dr. Carlos de Salles Soares Neto
Universidade Federal do Maranhão

Prof. Dr. Mário Antônio Meireles Teixeira
Universidade Federal do Maranhão

Prof. Dr. Tiago Bonini Borchartt
Universidade Federal do Maranhão

AGRADECIMENTOS

Bom, meu primeiro e maior agradecimento vai para aquele que sempre esteve comigo e sempre me ajudou nos piores momentos da minha vida, Deus. Obrigado por me dar forças em momentos que eu achei que não conseguiria, sou eternamente grato.

Em segundo queria agradecer à minha família que nunca me desmotivou em nada e sempre me apoiou em todas minhas escolhas. Obrigado mãe (Ana Cláudia), pai (Antônio José) e irmãos (Kleyton e Karyne) por comemorarem e se orgulharem de todas minhas conquistas e vitórias. Eu não escolheria jamais outra família para estar comigo, amo vocês!

Agora queria agradecer aos meus amigos que estiveram comigo do início ao final desta caminhada, que compartilharam comigo momentos de tensão, dúvidas e “sofrimento”, mas também de muita alegria. Agradeço ao C Squad: Saulo, Fernando, Lucas, Breno, João Leonardo e em especial Claudiny Priscila que esteve muito mais presente neste final de ciclo. Além deles também agradeço meu amigo Igor Estrela, Jorge e João Pedro, que sempre estiveram dando palavras de incentivo e motivação.

Agora queria agradecer em especial à dois professores que além de ótimos docentes, se tornaram grandes amigos meus. Meu orientador Carlos de Salles, que além de grande profissional, é um ótimo incentivador e amigo, além de ter um bom gosto para área nerd e geek. E o professor Mário Meireles que também sempre esteve me auxiliando e ajudando no que eu precisasse.

Por fim, eu não poderia deixar de agradecer quem ou o que me fez querer entrar no curso, o The King of Fighters 2002, sim, eu estou agradecendo um jogo. Esse jogo foi o motivador para entrar no curso e conseqüentemente me apaixonar por ele. Meu desejo era de desenvolver um igual, porém não aconteceu, entretanto, muitos outros foram possíveis e sou feliz por isso.

Agradeço também aos animes Cavaleiros do Zodíaco, Sword Art Online, Naruto, Dragon Ball, Digimon, Bleach, Pokémon, Shingeki no Kiojin, Kimetsu no Yaiba e outros que me ajudaram nos momentos que precisei relaxar e esquecer os problemas para que então pensasse em soluções para tais. Esses animes fizeram e sempre farão parte da minha vida, da minha história e por isso a gratidão.

Eu sou eternamente grato por tudo, e que esta gratidão perdure por muitos e muitos anos, pois é um sentimento que quero levar comigo até o fim dos meus dias.

“O caminho da sabedoria não é algo que se conquista, mas sim um aprendizado constante e eterno.”

(Shaka de Virgem – Cavaleiros do Zodíaco)

Lista de Ilustrações

Figura 1 – Quadro resumo do desenvolvimento por plataforma (Adaptado)	9
Figura 2 – Tela do Jogo Direção Defensiva (JDD)	10
Figura 3 – Tela de Início do Jogo Corona Quest	11
Figura 4 – Analogia à uma Construção para Representar as Tecnologias Web	13
Figura 8 – Jogo Assassin's Creed 4	16
Figura 9 – Jogo Dead Space	17
Figura 5 – <i>Framework</i> Phaser	18
Figura 6 – Exemplo de Modelo feito no Three.js	19
Figura 7 – Ambiente do melonJS	20
Figura 10 – Representação da Torre de Hanoi de Édouard Lucas (1842)	21
Figura 11 – Tela Inicial do Tower of Hanoi Game	22
Figura 12 – Tela Game do Tower of Hanoi Game	22
Figura 13 – Representação do puzzle do fazendeiro, o lobo, o carneiro e a alface	23
Figura 14 – Tela Inicial do Dangerous Crossing Game	23
Figura 15 – Tela Game do Dangerous Crossing Game	24
Figura 16 – Tela Inicial do Cosmo	24
Figura 17 – Funcionamento da Tela Inicial do Tower of Hanoi Game	26
Figura 18 – Modal de Créditos	26
Figura 19 – Modal de Informações do Dangerous Crossing Game	27
Figura 20 – Funcionamento da Tela Inicial do Tower of Hanoi Game	27
Figura 21 – Modo Clique do Tower of Hanoi Game	28
Figura 22 – Modo Parágrafo do Dangerous Crossing Game	29
Figura 23 – Diretório dos jogos	29
Figura 24 – Componente de Input	30
Figura 25 – Campos de digitação dos jogos	31
Figura 26 – Mascote dos jogos - Cosmolito	32
Figura 27 – HUD do Tower of Hanoi Game	32
Figura 28 – Mosaico com Modais de Alertas do Jogo	34
Figura 29 – Mosaico com Modais de Derrotas do Dangerous Crossing Game	34
Figura 30 – Representação de Alt em Imagens	35
Figura 31 – Jogo Tower of Hanoi Game Versão Mobile	36
Figura 32 – Tela Game do Framework 1.0	40
Figura 33 – Variações do Framework 1.0	40

RESUMO

O desenvolvimento de jogos sérios para a educação e tecnologias Web está crescendo. Da mesma forma, o desenvolvimento de jogos baseados na web tem ganhado espaço. Criar um *framework* personalizado para a criação desses jogos acelera o desenvolvimento e garante a qualidade do jogo. O *Game Design* desempenha um papel crucial na eficácia desses jogos, enquanto a usabilidade visa melhorar o aprendizado. A aplicação de elementos do *Game Design* e usabilidade nos jogos Tower of Hanoi Game e Dangerous Crossing Game mostraram pontos positivos após aplicação em sala de aula com os alunos.

Dessa forma, este trabalho busca apresentar a viabilidade da construção de um *framework* voltado para o desenvolvimento de jogos sérios Web para integrarem o conteúdo apresentado pelas disciplinas que visam o ensino de computação, e em contrapartida evidenciar um conjunto de diretrizes que auxiliem a comunidade de desenvolvedores de jogos sérios levando em consideração elementos visuais e estruturais incorporados nos jogos desenvolvidos e analisados no presente trabalho. Isso é feito por meio da checagem de aspectos do *game design* e usabilidade que ajudarão a melhorar a experiência dos alunos, e conseqüentemente o aprendizado.

Palavras-chave: Jogos Web, Jogos Sérios, Tecnologias Web, *Framework*, *Game Design*, Usabilidade.

ABSTRACT

The development of serious games for education and web technologies is growing. Similarly, web-based game development is gaining momentum. Creating a custom framework for building these games accelerates development and ensures game quality. Game Design plays a crucial role in the effectiveness of these games, while usability aims to enhance learning. The application of Game Design and usability elements in games like Tower of Hanoi Game and Dangerous Crossing Game showed positive results after implementation in the classroom with students.

Thus, this work aims to present the feasibility of constructing a framework focused on the development of web-based serious games to integrate content presented in computer science courses, and, in turn, to provide a set of guidelines that assist the community of serious game developers, considering visual and structural elements incorporated in the games developed and analyzed in this study. This is achieved through checking aspects of Game Design and usability that will help enhance the students' experience and, consequently, their learning.

Keywords: *Web Games, Serious Games, Web Technologies, Framework, Game Design, Usability.*

SUMÁRIO

1	INTRODUÇÃO	6
2	REFERENCIAL TEÓRICO.....	8
2.1	Jogos Web e Jogos Sérios.....	8
2.2	Tecnologias Web.....	11
2.3	<i>Game Design</i> e Usabilidade em Jogos Sérios Web.....	14
2.4	Heads-Up Display (HUD)	15
2.5	<i>Frameworks</i> para Jogos	17
2.5.1	Phaser	18
2.5.2	Three.js	19
2.5.3	melonJS	20
3	TOWER OF HANOI GAME E DANGEROUS CROSSING GAME.....	21
4	APLICAÇÕES	25
4.1	Metodologia do Tower of Hanoi Game e Dangerous Crossing Game.....	25
4.1.1	Funcionamento e Implementação dos jogos	25
4.1.2	Aplicação do <i>Game Design</i> e da Usabilidade	30
5	RESULTADOS.....	37
5.1.1	Diretrizes de <i>Game Design</i> e Usabilidade.....	37
5.1.2	Framework 1.0.....	39
6	CONCLUSÃO	42
	REFERÊNCIAS.....	45

1 INTRODUÇÃO

Os jogos de modo geral têm criado muito espaço dentro do campo do aprendizado, como auxiliador dentro das escolas ou universidades. No cenário de jogos, existe uma classificação intitulada Jogos Sérios e segundo Oliveira *et al.* (2022) estes jogos possuem centralmente o objetivo educacional, eles contribuem no ensino-aprendizagem e podem ser aplicados em inúmeras áreas do conhecimento. Estes jogos podem ser desenvolvidos através de diferentes motores e ferramentas, como por exemplo as tecnologias Web.

As aplicações Web têm ganho mais espaço sobre as aplicações tradicionais, que tem se tornado cada vez menos atrativas. A facilidade de uso e liberdade do usuário em relação às aplicações Web é uma das principais vantagens desse tipo de aplicação, dizem Santos e Battaiola (2006). E quando se refere a jogos para a Web, a não necessidade de instalação de aplicativo ou bibliotecas facilita o acesso do usuário, e esse proveito se torna mais óbvio, uma vez que as aplicações Web podem ser acessadas em qualquer parte do mundo, necessitando apenas de internet, ressaltam Santos e Battaiola (2006).

Para que essas aplicações/jogos sejam instigadores e atrativos, é necessário atentar-se para vários aspectos e elementos, como aqueles existentes no *Game Design*.

Além dos aspectos técnicos de um jogo sério, o *game design* de um jogo é uma ferramenta crucial para se alcançar a eficácia dentro dos jogos sérios, ainda mais em relação a jogos digitais que necessitam de motivadores atrelados em contrapartida com a aprendizagem, pelo teor instrucional, conforme Cezarotto e Battaiola (2017). Os modelos de *Game Design*, criados para instruir na concepção dos jogos, fornecem conjuntos de conceitos para auxiliar nesse desenvolvimento em todas as etapas do processo, reafirma Cezarotto e Battaiola (2017).

Assim como o *game design*, a usabilidade possui forte relevância na área de Interação Humano-Computador. A integração da usabilidade é um elemento essencial para o sucesso dos jogos sérios, especialmente quando se trata de garantir que os jogadores possam interagir de maneira eficiente e satisfatória com a experiência proposta. Guimarães *et al.* (2022) afirmam que a usabilidade é um atributo de qualidade que visa melhorar a eficácia do aprendizado por meio dos jogos sérios.

Santos, Pereira e Gonçalves (2013) mencionam que a ISO 9241-11 define a usabilidade como a medida em que um sistema, produto ou serviço pode ser utilizado por usuários específicos para atingir objetivos específicos também.

A usabilidade de produtos pode ser melhorada pela incorporação de características e atributos conhecidos como capazes de beneficiar os usuários em um contexto particular de uso. De modo a determinar o nível de usabilidade alcançado é necessário medir o desempenho e satisfação dos usuários trabalhando com um produto. (ISO 9241-11).

A utilização das tecnologias Web e a aplicação dos conceitos mencionados anteriormente permitem o desenvolvimento rápido de aplicações Web modernas, adaptáveis e reutilizáveis, como é o caso dos *frameworks*.

Um *framework* é um conjunto de ferramentas, bibliotecas e padrões que simplifica e agiliza o processo de desenvolvimento. Ao criar um *framework* voltado para jogos sérios, é possível ter à disposição um conjunto de recursos pré-definidos, como componentes, funcionalidades e boas práticas, que aceleram o desenvolvimento e garantem a consistência e qualidade dos jogos produzidos.

Considerando o contexto apresentado, o objetivo principal deste trabalho é apresentar a viabilidade da construção de um *framework* voltado para o desenvolvimento de jogos sérios Web a serem aplicados no ensino de computação, fazendo uso de elementos do *game design* e usabilidade que ajudarão a melhorar a experiência dos alunos, e conseqüentemente o aprendizado. Além disso, apresentar um conjunto de diretrizes norteadoras que auxiliem os desenvolvedores de jogos sérios levando em consideração os elementos visuais, estruturais e usuais incorporados nos jogos sérios Web Tower of Hanoi Game e Dangerous Crossing Game.

Este trabalho encontra-se organizado da seguinte forma: na Seção 2 são apresentados os referenciais teóricos, dando embasamento para o desenvolvimento e os resultados do presente estudo. Na Seção 3 são apresentados os jogos desenvolvidos (funcionamento e implementação) nesse estudo. Na Seção 4 são apresentados os resultados obtidos no presente trabalho. Na Seção 5 é apresentado a conclusão do trabalho. E por fim as referências deste estudo.

2 REFERENCIAL TEÓRICO

2.1 Jogos Web e Jogos Sérios

Segundo Santana, Fortes e Porto (2016) o jogo em si trata-se de uma atividade que tenha no mínimo um indivíduo praticante e regras estabelecidas, além disso eles são utilizados com intuito recreativo e estimulação física e mental. Ainda nesse contexto, Huizinga (2001) esclarece:

[...] o jogo é mais do que um fenômeno fisiológico ou um reflexo psicológico. Ultrapassa os limites da atividade puramente física e biológica. É uma função *significante*, isto é, encerra um determinado sentido. No jogo existe uma coisa “em jogo” que transcende as necessidades imediatas da vida e confere um sentido à ação. Todo jogo significa alguma coisa. Não se explica nada chamando “instinto” ao princípio ativo que constitui a essência do jogo [...] o simples fato de o jogo encerrar um sentido implica a presença de um elemento não material em sua própria essência.

Com o passar dos anos, a maneira de interagir com os jogos se alterou gradativamente, principalmente pelo advento da tecnologia, ressaltam Santana, Fortes e Porto (2016). Dessa forma as brincadeiras e os jogos de rua foram aos poucos se transformando em produtos do digital, atualmente conhecidos como jogos digitais.

Os jogos digitais são jogos que dependem de um dispositivo eletrônico para que o mesmo funcione, como exemplo computadores, máquinas e celulares. Esses dispositivos podem fazer o jogo ser executado por meio de fitas, cartuchos, discos, *internet* ou armazenamento de dados.

Araújo Júnior (2009) esclarece que os jogos web, diferente dos jogos de computador, rodam diretamente no navegador e não precisam ser instalados no computador do usuário. O que será necessário para que essa categoria de jogos funcione é a presença de internet e um navegador.

Em uma pesquisa desenvolvida em 2022 pela AbraGames (Associação Brasileira das Desenvolvedoras de Jogos Eletrônicos) com o objetivo de mapear os estúdios desenvolvedores de games e o crescimento do desenvolvimento de games nos últimos anos, tendo 157 desenvolvedores respondentes em relação a distribuição de jogos desenvolvidos para diferentes plataformas, nos últimos anos em resumo, obteve-se 13% de jogos desenvolvidos para plataforma web, como mostra a Figura 1.

Figura 1 – Quadro resumo do desenvolvimento por plataforma (Adaptado)



Fonte: AbraGames 2022

Conforme Biurrun (2019), os jogos Web ganharam mais popularidade especialmente pela facilidade de acesso e segurança, já que não é necessário a realização do download e muito menos a instalação dos jogos, pois os principais navegadores estão disponíveis para diversos sistemas operacionais. Isso torna os jogos Web multiplataformas, permitindo sua execução tanto no Windows quanto no Linux, sem a necessidade de configurações adicionais. Além disso, os navegadores garantem segurança, pois restringem o acesso dos jogos aos seus próprios escopos, impedindo que leiam arquivos pessoais dos usuários.

Essa facilidade e segurança é muito mais interessante quando é necessário a aplicação em sala de aula, visto que não haverá perda de tempo com instalação e isso não prejudicará o andamento da aula a qual o jogo está sendo aplicado, neste caso os jogos sérios.

Segundo Chiarato *et al.* (2018) os jogos sérios podem ser aplicados para que se alcance soluções para problemas em várias áreas que ainda estão em progresso, fazendo uso apenas como estratégia, o desafio a jogadores com os mais níveis de conhecimento, e ainda reforça que os jogos sérios podem proporcionar através do seu aprendizado um treinamento, desenvolvimento ou aprimoramento de alguma habilidade.

Muitos podem achar que as palavras entretenimento e educação podem se contradizerem, informa Cuba (2009), entretanto, isso não acontece, mas sim lugares específicos no contexto em que os dois se sobressaem e podem usar suas ferramentas para alcançar seus objetivos.

Nesse cenário os jogos sérios utilizam o entretenimento para veicular propósitos e alcançar a propagação do aprendizado ao jogador, como já mencionado seu objetivo central é o educacional. Um exemplo desta aplicação é o jogo Direção

Defensiva¹, Figura 2, que visa ensinar sobre as regras de trânsito e a direção segura no trânsito de forma interativa e marcante.

Ele foi desenvolvido pela PlayerUm², uma empresa que oferece soluções digitais, inovadoras e gamificadas por meio de plataformas interativas, jogos e aplicativos.

Figura 2 –Tela do Jogo Direção Defensiva (JDD)



Fonte: Play Store – Google Play

Ainda dentro deste cenário, outro exemplo de jogo sério é o CoronaQuest³. Desenvolvido em 2020 pelo Departamento de Educação e Treinamento, Juventude e Cultura (DFJC) de Cantão de Vaud na Suíça, foi ganhador de vários prêmios, incluindo o 1º lugar no International Educational Games Competition 2021⁴ na categoria de *Web-based Games*.

Ele é um jogo de cartas (Figura 3) que visa conscientizar crianças sobre as medidas a serem tomadas para preservar sua saúde e como lidar com o coronavírus tanto na escola como em outros ambientes.

¹ https://play.google.com/store/apps/details?id=com.Playerum.JogoDirecaoDefensiva&hl=pt_BR

² <https://playerum.com.br>

³ <https://coronaquest.game>

⁴ <https://www.academic-conferences.org/conferences/ecgbl/ecgbl-future-and-past/>

Figura 3 – Tela de Início do Jogo Corona Quest



Fonte: Corona Quest

2.2 Tecnologias Web

Segundo Biurum (2019) para o desenvolvimento de sistemas para Web, o mais aconselhável de se usar seriam as tecnologias: *Hypertext Markup Language* (HTML), *Cascading Style Sheets* (CSS), React e JavaScript/TypeScript, cada uma desempenha uma função singular quando se refere a fazer o jogo “existir”.

Para evitar grande parte do trabalho, em sua maioria os desenvolvedores de jogos buscam a utilização de motores de desenvolvimento de jogos digitais, contudo, a união de forma inteligente das tecnologias apresentadas acima, possibilitará na criação de qualquer tipo de jogo para a internet. Por esse motivo será necessário esmiuçar sobre cada uma dessas tecnologias abordadas neste trabalho.

Segundo Caldeira (2015) o HTML trata-se de um conjunto estruturado de instruções que são conhecidas por etiquetas ou *tags*, que indicam ao navegador como apresentar uma página Web, ou seja, o navegador faz a interpretação dessas *tags* e então desenha a página no visor.

A estrutura básica de um sistema em HTML é composta pelas seguintes *tags*:

- ✓ **<html></html>**: Marca o início e fim da página Web;
- ✓ **<head></head>**: É o cabeçalho do sistema, é nele que se coloca os elementos que não aparecem na página;

- ✓ **<title></title>**: É onde se coloca o título da página que aparece na aba do navegador;
- ✓ **<body></body>**: É onde é colocado todo o conteúdo que deverá aparecer na visor, ou seja, a página Web.

Vale ressaltar que essa é a estrutura básica, logo, existem inúmeras outras *tags* que contribuem para a criação e finalização do sistema Web.

O CSS, segundo Ribeiro *et al.* (2006) permite fazer coisas que não era possível somente utilizando o HTML, como por exemplo delimitar margens, especificação de fontes, *backgrounds*, cores e muitas outras modificações.

O CSS pode ser escrito dentro do próprio HTML ou em um arquivo separado (externo) e então ser “chamado” na *tag* `<head></head>` do HTML. O que era feito em várias linhas de código, hoje se tornou mais leve em tamanho de *bytes*, pois a estrutura é bem menor, ressalta Ribeiro *et al.* (2006).

Conforme Neves (2023) o React é uma biblioteca que foi desenvolvido pela Facebook para criar interfaces de usuário em aplicativos Web. E logo ganhou espaço por ser fácil de usar e altamente flexível e escalável.

Um dos principais casos de uso dessa biblioteca é criar aplicações Web que precisam ser atualizadas em tempo real. Neves (2023) diz que o React é uma biblioteca modular, ou seja, os componentes (blocos de construção) podem ser facilmente reutilizados e compartilhados em diferentes momentos e partes do sistema/aplicação.

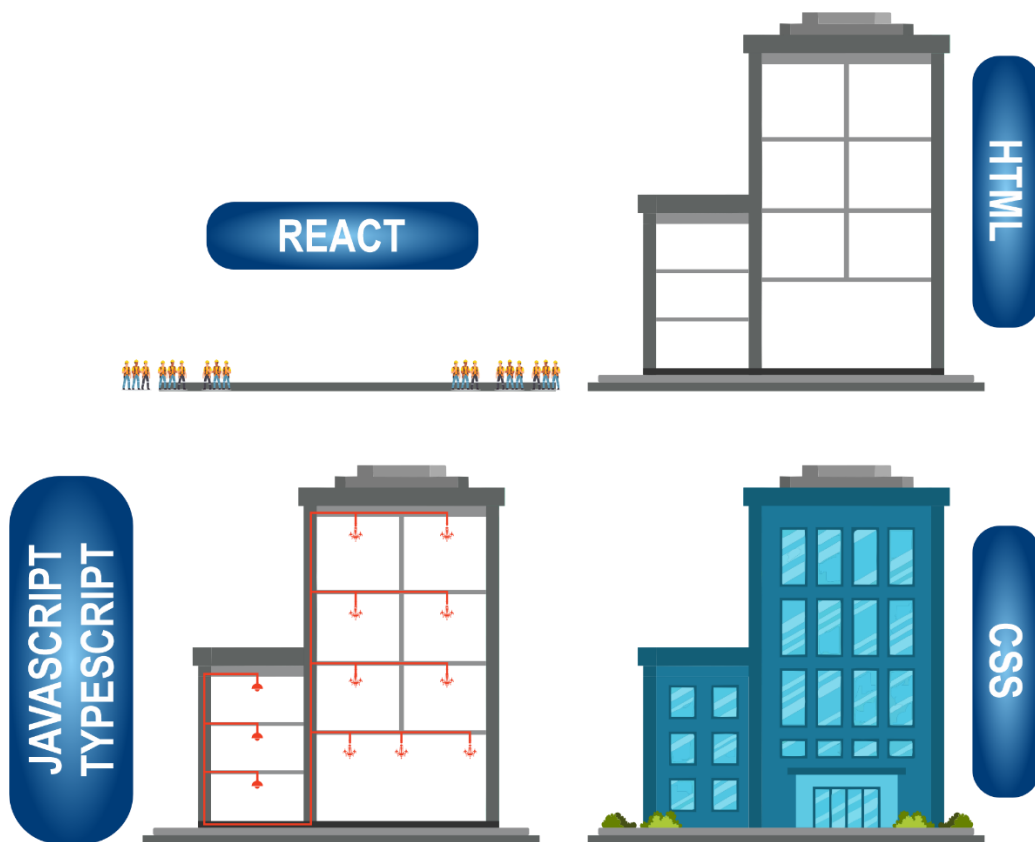
Por fim, tem-se o JavaScript e o TypeScript que segundo Nascimento (2021) esse primeiro é uma linguagem de programação dinâmica, flexível e tipada. O TypeScript é também uma linguagem de programação e um superconjunto do JavaScript, ou seja, todo o código que seja escrito em JavaScript é perfeitamente aceito pelo TypeScript, entretanto, não no sentido contrário.

A principal diferença entre os dois é a tipagem dos dados, o uso das classes e os mecanismos de abstração. Na documentação do TypeScript, é destacado que “O benefício principal do TypeScript é que ele pode destacar comportamento inesperado no seu código, diminuindo a chance de bugs” (TYPESCRIPT, 2012).

Para que se entenda melhor a combinação de todas essas tecnologias, é possível fazer uma analogia à um projeto de construção, e assim ter mais claramente o papel de cada um dentro do sistema, conforme a Figura 4.

O HTML por exemplo é responsável por ser o esqueleto de um edifício, ele estrutura as coisas e define a hierarquia dos elementos; o CSS por sua vez, seria a pintura e o design de interiores de um edifício, ou seja, ele adiciona a estilo e aparência visual, deixa algo atraente e esteticamente agradável; o JavaScript/TypeScript pode ser comparado com o sistema elétrico ou a automação do edifício, pois ele oferece funcionalidades interativas, dinâmicas e controle sobre o ambiente, e por fim, o React se trata de uma equipe de construção altamente eficiente, pois atua na simplificação do processo de construção, reutiliza materiais e facilita na manutenção.

Figura 4 – Analogia à uma Construção para Representar as Tecnologias Web



Fonte: Elaborado pelo autor

Para que um jogo seja executado na web, é necessário que o computador tenha inicialmente dois elementos importantes – internet e navegador (browser) – segundo Mozelli (2023), os navegadores mais utilizados entre os usuários no computador são Google Chrome, Apple Safari, Microsoft Edge, Mozilla Firefox, e Opera, e conforme as tecnologias web vão avançando e melhorando, os navegadores também seguem esse curso de aprimoramento.

2.3 *Game Design* e Usabilidade em Jogos Sérios Web

Cezarotto e Battaiola (2017) destacam a importância de considerar não apenas o jogo em si, mas também o *gameplay* e sua relação com o currículo escolar ao abordar o *game design* para jogos sérios. Para alcançar um aprendizado efetivo, vários elementos são analisados, como sons, animações e outros elementos (CEZAROTTO E BATTAIOLA, 2017).

Além disso, ao realizar um *Cognitive Walkthrough* (passo a passo cognitivo), segundo Kieras (2006) é possível avaliar se o usuário consegue reconhecer que suas ações produzem os efeitos desejados, e se o *feedback* fornecido permite ao usuário saber que teve sucesso após tomar determinada ação. O *Cognitive Walkthrough* é uma técnica para avaliar a capacidade de um sistema de aprendizado sob a perspectiva de um usuário iniciante. Em contraste com a avaliação de usuário, ele prescinde da participação direta de usuários.

Segundo Kieras (2006), para que se tenha um *design* efetivo é preciso que se evite uma tela com muitas informações e/ou muito “poluída”, pois com um bom *layout* é possível apresentar efetivamente muito mais informações do que se pensa.

A interface dos jogos sérios, também conhecida como *Heads-Up Display* (HUD), tem como objetivo transmitir informações e *feedbacks* ao usuário (Tóneis, 2012). Essa interface é considerada o principal meio de comunicação com o jogo, portanto, sua construção deve priorizar clareza e objetividade.

Quanto à usabilidade, para que se qualifique quão bem o usuário consegue interagir com o sistema interativo, nesse caso específico, com os jogos sérios, Guimarães *et al.* (2022) citam em seu trabalho Nielsen (1993) que define cinco fatores de usabilidade: facilidade de aprendizado, segurança no uso, facilidade de recordação, eficiência de uso e satisfação do usuário. Esses fatores são essenciais para avaliar a interação do usuário com o sistema interativo.

Com objetivo de atender esses fatores, Oliveira Júnior, Ferreira e Marques (2020) em seu trabalho citam que Juristo *et al.* (2007) apresentam as *Functional Usability Features* (FUFs), que trata de um conjunto de aspectos funcionais de usabilidade referentes ao impacto no *design* e arquitetura do sistema. Os autores mencionam que esse conjunto pode ser decomposto em mecanismos, nos quais um ou mais de um deles podem ser implementados em um *software*, não

necessariamente todos. O Quadro 1 apresenta os mecanismos apontados por Juristo *et al.* (2007).

Quadro 1 – Mecanismos de Usabilidade (Criado a partir de Juristo *et al.* (2007))

Status do Sistema: Informa o status interno do sistema.	Voltar: Retorna à um determinado estado de execução.
Interação: Informa que o sistema realizou o registro de uma interação.	Entrada de Texto Estruturada: Preveni cometer erros de entrada de dados.
Alerta: Informa sobre qualquer ação com consequências importantes.	Execução Passo a Passo: Auxilia em tarefas que necessitam de diferentes passos com entrada de dados correta.
Feedback sobre Progresso: Informa se irá demorar para concluir processamento de ação.	Preferências: Registra ações no uso das funções do sistema.
Desfazer: Desfaz várias ações em um objeto.	Área de Objetos Pessoais: Registra as opções no uso da interface do sistema.
Cancelar: Cancela a execução de uma ação.	Favoritos: Registra partes do sistema e o conteúdo que são de interesse.
Abortar Operação: Cancela a execução de todo o sistema.	Ajuda Multinível: Disponibiliza diferentes níveis de ajuda.
Agregação de Comando: Disponibiliza um conjunto de comandos para acessar os recursos do sistema.	

Fonte: Juristo *et al.* (2007)

Outros fatores são considerados para se obter um maior direcionamento para os engenheiros de *software*, entretanto, esse trabalho se atentará apenas aos aspectos já mencionados anteriormente, o *game design* e a usabilidade do sistema.

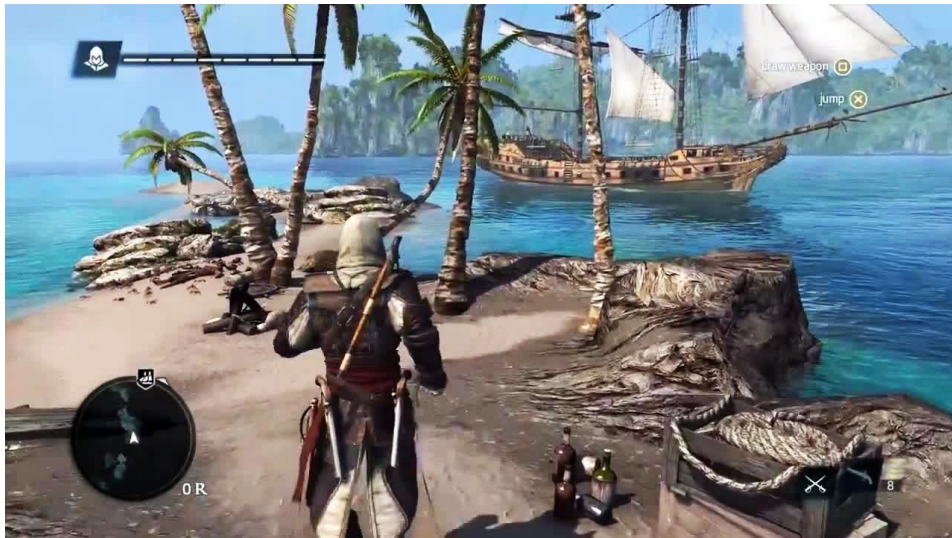
2.4 Heads-Up Display (HUD)

Todo sistema que têm regras necessita de suportes estatísticos e *feedbacks* que auxiliem na demonstração de progresso ou ações, no caso dos jogos em especial a informação constante para o jogador de sua situação atual no jogo, ressaltam Maroni e Battaiola (2011).

A HUD é a interface gráfica que interage com o jogador e o informa questões importantes sobre seu progresso ou desenvolvimento no jogo, existem vários tipos de HUD, mas este trabalho se atentará a apenas dois. Segundo Maroni e Battaiola (2011) as informação em um jogo podem se dar por meio de barras medidoras, caixas de

texto, e outros conteúdos. Essa interface se diferencia do mundo do jogo em sua maioria e se encontra em um plano totalmente diferente. Um exemplo desse tipo de HUD é presenciada no jogo Assassin's Creed 4, como mostra a Figura 8.

Figura 5 – Jogo Assassin's Creed 4



Fonte: Youtube

Esse tipo de HUD segue a tendência dos jogos mais usuais da década, que acaba por manter uma aparência limpa, mostrando apenas o necessário na tela, como por exemplo a bússola no canto inferior esquerdo e a barra de vida no canto superior esquerdo.

Ao longo do desenvolvimento de novas técnicas e possibilidades no cenário dos jogos, alguns novos caminhos foram sendo trilhados, e alguns jogos já buscam que sua HUD seja integrada ao mundo do jogo, traduzindo como sinais que sejam subjetivos ou até mesmo objetivos dentro do mundo e assim indiquem a situação para o jogador, dizem Maroni e Battaiola (2011).

Tóneis (2012) ainda cita que vários pesquisadores identificaram e denominaram essa HUD de Interface Aplicada ao Mundo de jogo, e o seu principal ponto é a integração da mesma com o cenário do jogo. Um exemplo dessa tipo de HUD é o jogo Dead Space (Figura 9), pois uma das características é quando o contador de munição é projetado sobre cada arma quando o personagem está realizando a mira.

Figura 6 – Jogo Dead Space



Fonte: Persona – Jornalismo Cultural

Segundo Maroni e Battaiola (2011), a HUD de um jogo digital deve comunicar corretamente todas as informações e ainda citam que Santos *et al.* (2007) ressalta que uma interface ideal deve se apoiar em três princípios dentro dos jogos:

- ✓ Jogabilidade: A forma que o jogador interage com o jogo;
- ✓ Funcionalidade: A variedade de funções que é possível executar no jogo; e
- ✓ Facilidade de Compreensão: Uma interface recém-apresentada ao jogador deve ser de fácil compreensão.

2.5 Frameworks para Jogos

Cavalcante (2021) cita que Johnson (1997) define *framework* como um *design* de todo ou parte de um sistema que é representado por um conjunto de classes abstratas. Outra definição expressa por Johnson (1997) é que um *framework* se trata de um esqueleto de uma aplicação que pode ser customizada por um desenvolvedor de aplicativos/sistemas. Segundo Pereira (2017), os *frameworks* existem com objetivo de minimizar a codificação de problemas que já foram resolvidos e ainda maximizar a reutilização de códigos que já foram testados.

Pereira (2017) ressalta que para um *frameworks* funcionar eficientemente precisa ter funções e métodos muito bem definidos, são eles: Inicializar (responsável

por iniciar todos os recursos necessários para o desenvolvimento do jogo); Carregar (carrega os recursos tais como, imagens, vídeos, sprites dos personagens, áudio, cenários etc.); Atualizar (insere toda a lógica do jogo que será constantemente atualizado dentro do looping do jogo); Renderizar (chama 60 vezes por segundo todos os objetos desenhados dentro do método) e por fim Descarregar (descarrega todos os recursos não utilizados no método de “Atualizar” e “Renderizar”), e Apesar de algumas diferenças entre as diversas frameworks, em geral, todas possuem conceitos semelhantes (PEREIRA, 2017).

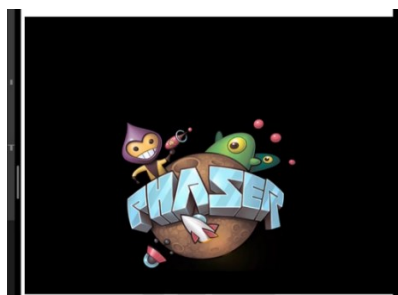
Os benefícios de se utilizar *frameworks* segundo Cavalcante (2021) estão entre a modularidade, pois reduz o esforço para se entender e manter o *software*, a reutilização como já mencionado, já que evita a recriação e a revalidação de soluções para os requisitos do *software*, e a extensibilidade já que cria a possibilidade de personalização de novos recursos nas aplicações.

Esses benefícios são de fundamental relevância quando se refere à produção de jogos sérios para a inserção no currículo de uma determinada disciplina. Alguns *frameworks* existentes são usados para a produção de jogos digitais no contexto geral e não especificamente sérios, abaixo segue a apresentação de alguns destes.

2.5.1 Phaser

O Phaser⁵ (Figura 5) é um *framework* em HTML5 para o desenvolvimento de jogos que rodem nas plataformas desktop e mobile. Com o aumento do desempenho e do uso dos navegadores, ele foi criado utilizando as tecnologias Web.

Figura 7 – Framework Phaser



Fonte: Design Code

Segundo Silva *et al.* (2019) este *framework* permite ao desenvolvedor realizar o controle dos mais diferentes elementos presentes em um jogo, como personagens,

⁵ <https://newdocs.phaser.io/docs/3.60.0>

cenário, física e isso permite concentrar-se em aspectos específicos da criação, em vez de se preocupar com a configuração da lógica que define a mecânica do jogo.

2.5.2 Three.js

Segundo Danchilla (2012) Three.js⁶ é um *framework* JavaScript que a criação de modelo 3D para a Web, como mostra a Figura 6.

Esse *framework* dispõe de algumas características, como:

- ✓ Adaptação suave para o contexto 2D caso o WebGL não seja suportado;
- ✓ Incorporação de operadores para manipulação de vetores e matrizes;
- ✓ Utilitários para importação e exportação de recursos;
- ✓ Disponibilidade de uma documentação abrangente e diversos exemplos ilustrativos.

Three.js é um *framework* JavaScript que a criação de modelo 3D para a Web, diz Danchilla (2012).

Figura 8 – Exemplo de Modelo feito no Three.js



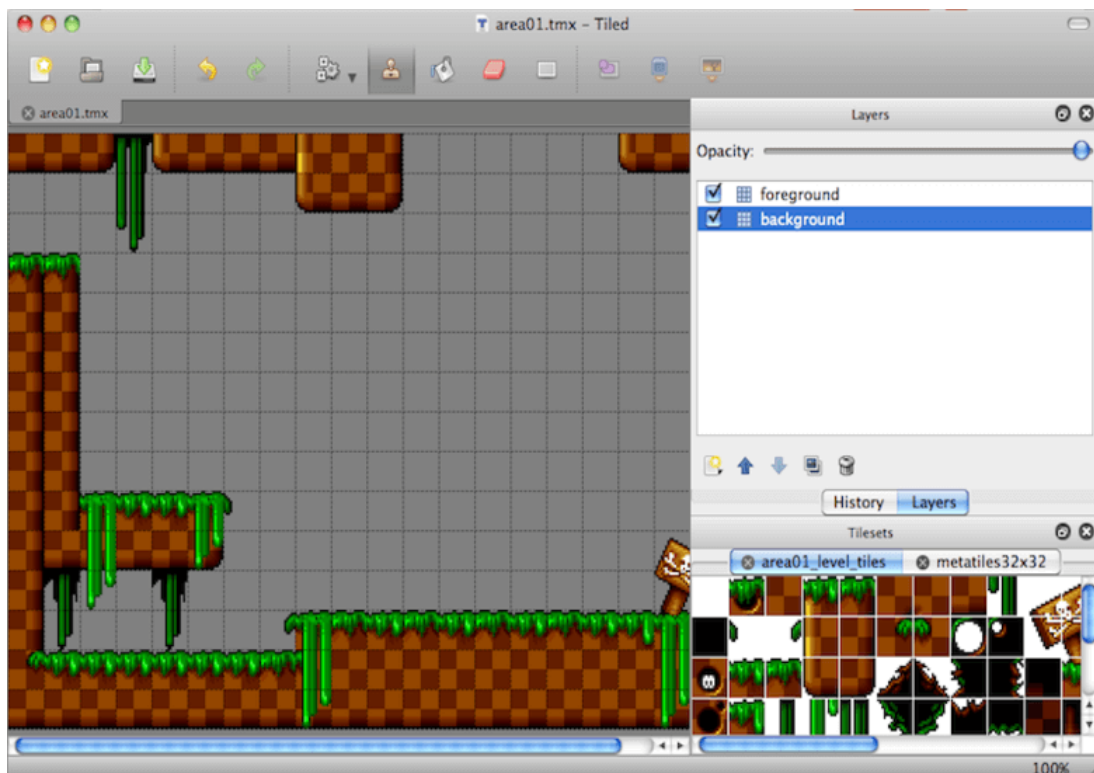
Fonte: Three.js

⁶ <https://threejs.org>

2.5.3 melonJS

O melonJS⁷ (Figura 7) é um *framework* de código aberto que trabalha em um contexto 2D. Ele possui um formato de mapa popular que permite criar níveis de forma fácil usando um editor de mapas, o que dá ao desenvolvedor mais tempo para se concentrar na implementação das características específicas do jogo em desenvolvimento, afirma Lopes e Silveira (2015).

Figura 9 – Ambiente do melonJS



Fonte: Produção de Jogos

A construção de jogos com esse *framework* é baseada no uso de recursos modernos e não requer dependência de outras bibliotecas. Além disso, o melonJS tem gerenciamento de animações e efeitos incorporados (como tingimento e máscaras), além de efeitos de transição e transição suave, e ainda é compatível com diversos navegadores Web do mercado, incluindo os de plataforma *mobile*.

⁷ <https://melonjs.org>

3 TOWER OF HANOI GAME E DANGEROUS CROSSING GAME

Segundo Lima (2013), A Torre de Hanoi, também conhecida como torre de bramanismo ou quebra-cabeças do fim do mundo, foi inventada e vendida como um brinquedo em 1883 pelo francês matemático Edouard Lucas, como na Figura 10. O francês se inspirou em uma lenda Hindu a qual falava de um templo em Benares. uma cidade Santa na Índia, na qual existia uma torre sagrada do bramanismo, cuja função era melhorar a disciplina mental dos jovens monges.

Figura 10 – Representação da Torre de Hanoi de Édouard Lucas (1842)



Fonte: University of Evansville - Mike Ciholas

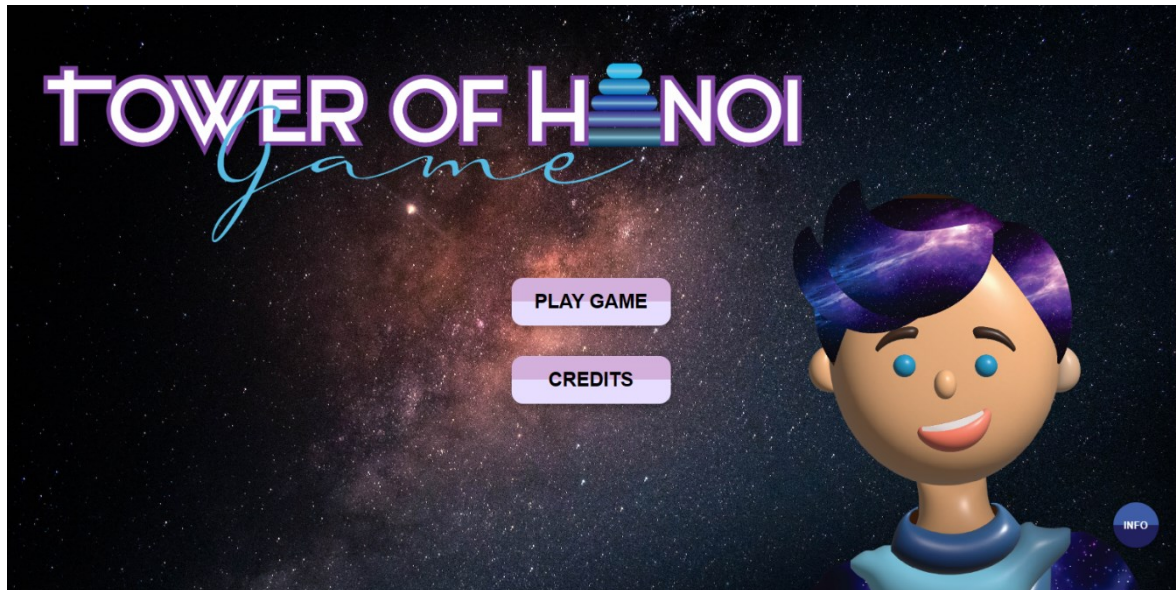
Segundo a lenda, os monges foram incumbidos de um desafio: transferir a torre composta por discos de uma haste para outra, contando com a ajuda de uma terceira haste como auxiliar. No entanto, havia restrições específicas que deveriam ser seguidas rigorosamente: eles só poderiam mover um disco de cada vez e nunca colocar um disco maior sobre um menor.

Para completar essa tarefa, os monges precisavam trabalhar com eficiência, dia e noite, sem interrupções. Ao término do trabalho, o destino do templo seria selado, transformando-se em pó, resultando no colapso do mundo.

A Torre de Hanoi tem como regras e objetivos de modo geral, transferir todos os discos que se encontram no pino inicial, para o ultimo pino com o mínimo de movimentos possíveis, movendo um disco de cada vez e nunca permitindo que se tenha um disco maior sobre um disco menor, explica Lima (2013).

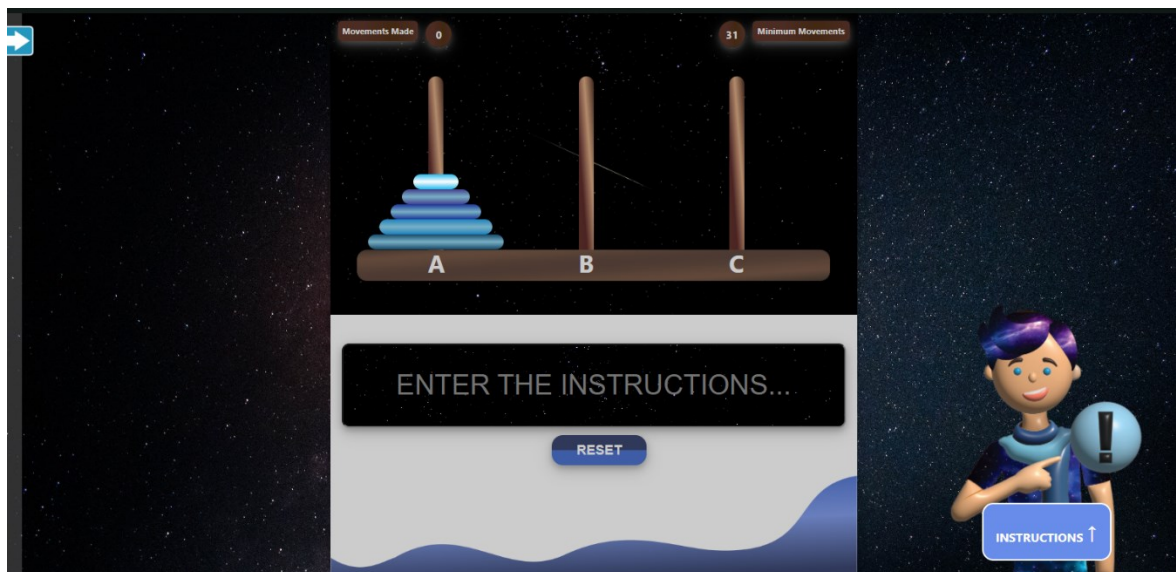
Com base no jogo/brinquedo original se desenvolveu o Tower of Hanoi Game, caracterizado com as mesmas peculiaridades do original e com as mesmas regras, só que em versão digital, como mostra a Figura 11 e a Figura 12.

Figura 11 – Tela Inicial do Tower of Hanoi Game



Fonte: Elaborado pelo autor

Figura 12 – Tela Game do Tower of Hanoi Game



Fonte: Elaborado pelo autor

O segundo jogo – Dangerous Crossing Game, foi inspirado em um clássico e famoso puzzle do século IX – problema do fazendeiro, o lobo, o carneiro e a alface – que por se tratar de uma expressão cultural, existem inúmeras versões conforme diz Ascher (1990). Esse puzzle conta que o fazendeiro precisa atravessar um rio levando um lobo, um carneiro e uma alface, como mostra Figura 13, mas só pode levar consigo uma das compras de cada vez.

Se deixar o lobo sozinho com o carneiro, o lobo o comerá. Se deixar o carneiro sozinho com a alface, o carneiro a comerá. O objetivo é atravessar o rio sem que nenhuma das compras seja prejudicada.

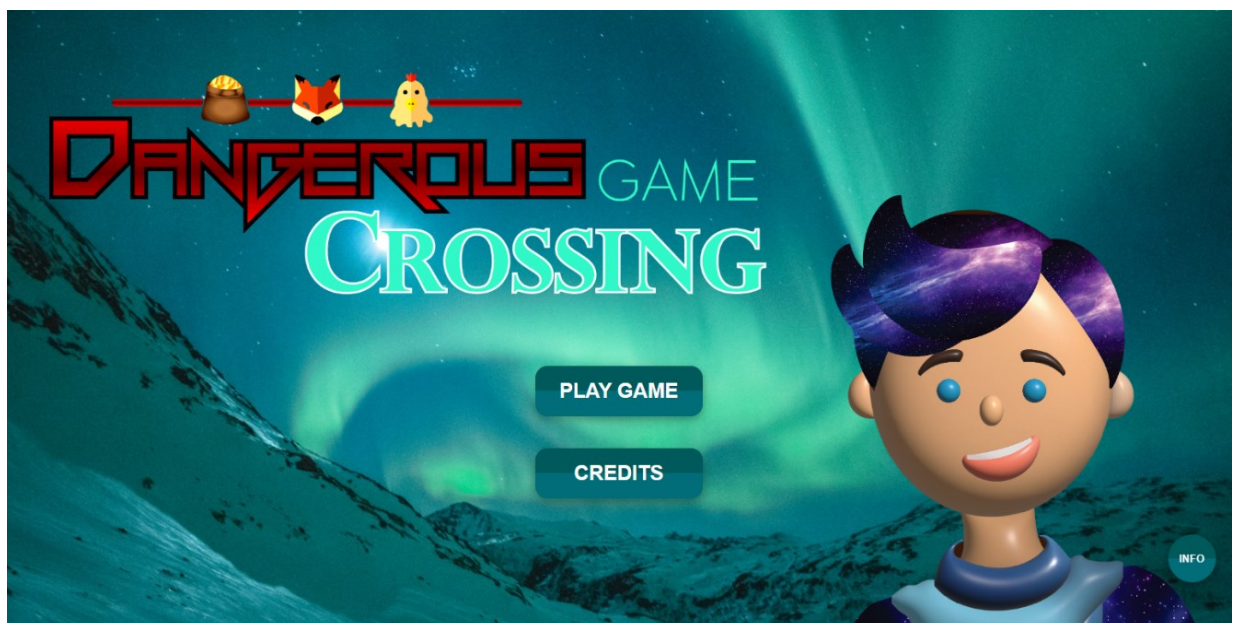
Figura 13 – Representação do puzzle do fazendeiro, o lobo, o carneiro e a alface



Fonte: Cátia Osório

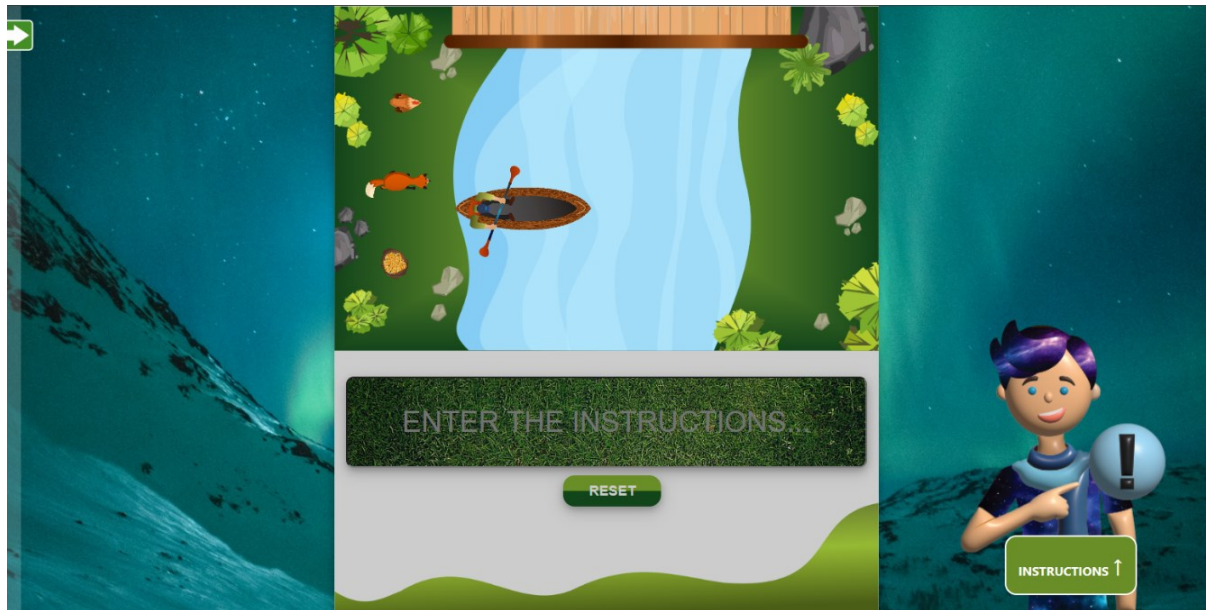
O Dangerous Crossing, em sua versão, utilizou um barqueiro, uma raposa, uma galinha e um saco de milho como os personagens e consiste em modo geral, na mesma regra mencionada no original. O aluno ao acessar o jogo tem a tela inicial bem semelhante ao jogo anterior, com mudanças apenas nas cores, logo e no plano de fundo, como mostra Figura 14 e a Figura 15.

Figura 14 – Tela Inicial do Dangerous Crossing Game



Fonte: Elaborado pelo autor

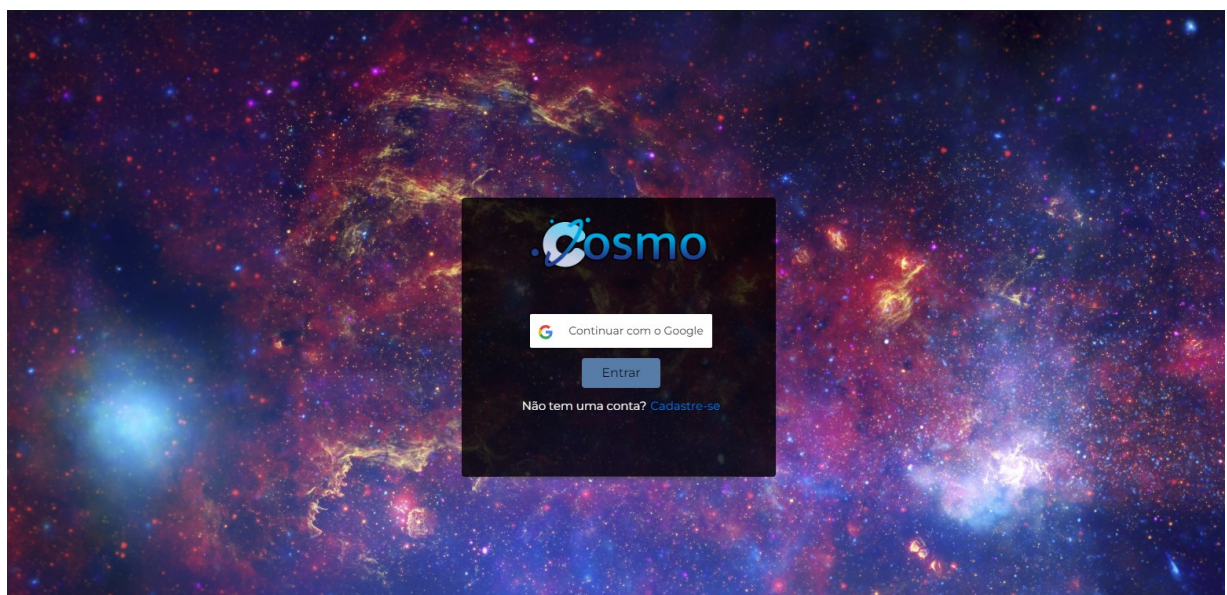
Figura 15 – Tela Game do Dangerous Crossing Game



Fonte: Elaborado pelo autor

Ambos os jogos foram desenvolvidos para integrarem no futuro o Cosmo⁸, Figura 16, uma plataforma de ensino, multitarefa focado em atividades voltadas ao estudo de uma disciplina de Introdução a Algoritmos, utilizada no primeiro período do Curso de Ciência da Computação da Universidade Federal do Maranhão – UFMA, por este motivo os jogos foram desenvolvidos separadamente da plataforma.

Figura 16 – Tela Inicial do Cosmo



Fonte: Elaborado pelo autor

⁸ <https://cosmo.telemidia-ma.com.br>

4 APLICAÇÕES

Esta seção visa apresentar os principais elementos, peculiaridades e funcionamento dos jogos sérios já apresentados, bem como mostrar as escolhas feitas na implementação para que fossem utilizados na disciplina de algoritmos I, com intuito de familiarizar os alunos com a sintaxe de programação.

4.1 Metodologia do Tower of Hanoi Game e Dangerous Crossing Game

Nesta seção são apresentados o funcionamento dos jogos Tower of Hanoi Game e Dangerous Crossing Game, a estrutura geral dos jogos, implementação, e a aplicação de elementos de *Game Design* e Usabilidade. A Seção 5.1.1 apresenta como os jogos funcionam e detalhes da sua implementação. Já na Seção 5.1.2 são apresentados os elementos de *Game Design* e Usabilidade aplicados nos jogos, que serão fundamentais para o entendimento das diretrizes presentes na Seção 6.

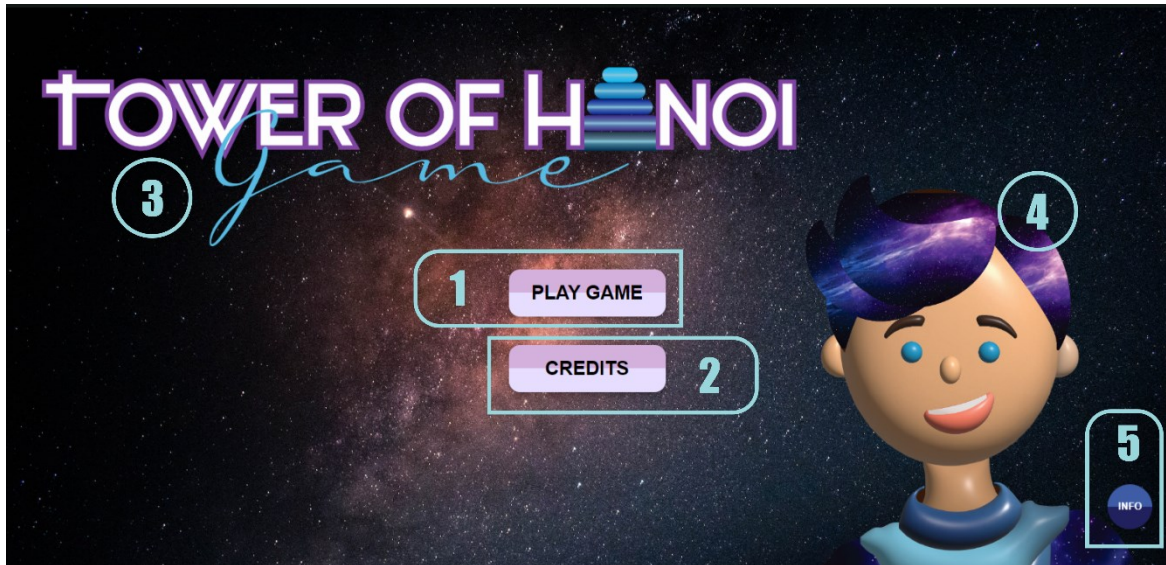
4.1.1 Funcionamento e Implementação dos jogos

Como já mencionado anteriormente, os jogos foram criados para integrarem o Cosmo, e por este motivo em especial o jogo Tower of Hanoi Game foi desenvolvido e estilizado com elementos que remetem cosmos ou galáxias com o objetivo de padronização e harmonização com o ambiente do Cosmo. Como ambos os jogos foram desenvolvidos seguindo os mesmos padrões estruturais, visto que visam a criação de um framework, nesta seção a exemplificação ocorrerá de forma aleatória entre os jogos.

Sobre o funcionamento, assim que o aluno acessa o jogo Tower of Hanoi Game, ele se depara com a tela inicial, como mostra a Figura 17. No marcador 1, existe o botão “*Play Game*” para iniciar o jogo; no marcador 2, o botão de “*Credits*”, que abre um modal com as informações acerca dos autores do jogo, conforme Figura 18; no marcador 3 têm a logo oficial do jogo de autoria do mesmo autor deste trabalho; no marcador 4 têm o mascote do jogo de nome Cosmolito, também desenvolvido pelo autor deste trabalho, e ao passar o mouse pelo personagem, ele profere a seguinte frase para o usuário de boas-vindas: “*Hi, my name is Cosmelito. Welcome to my game Tower of Hanoi. Have fun my young grasshopper. If you want to know more, click on*

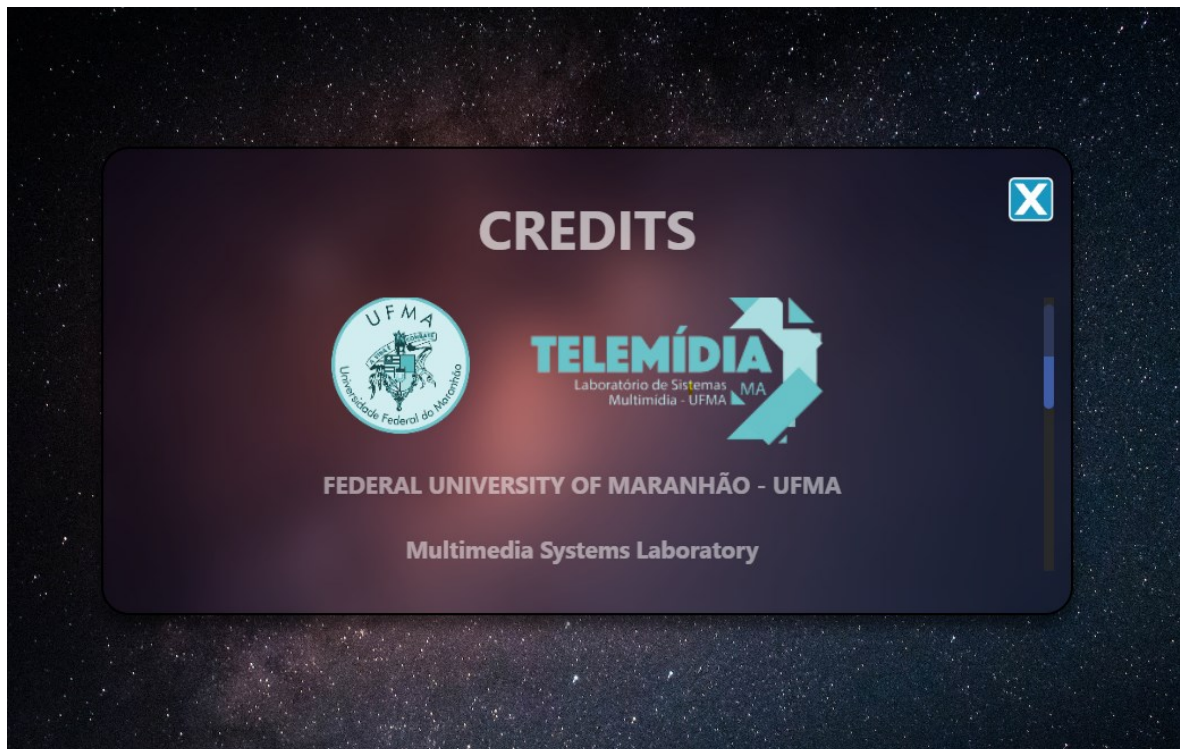
the floating button.”, e por fim, no marcador 5 tem o botão “Info”, que ao clicar abre um modal que contém informações gerais sobre A Torre de Hanoi, como mostra a Figura 19.

Figura 17 – Funcionamento da Tela Inicial do Tower of Hanoi Game



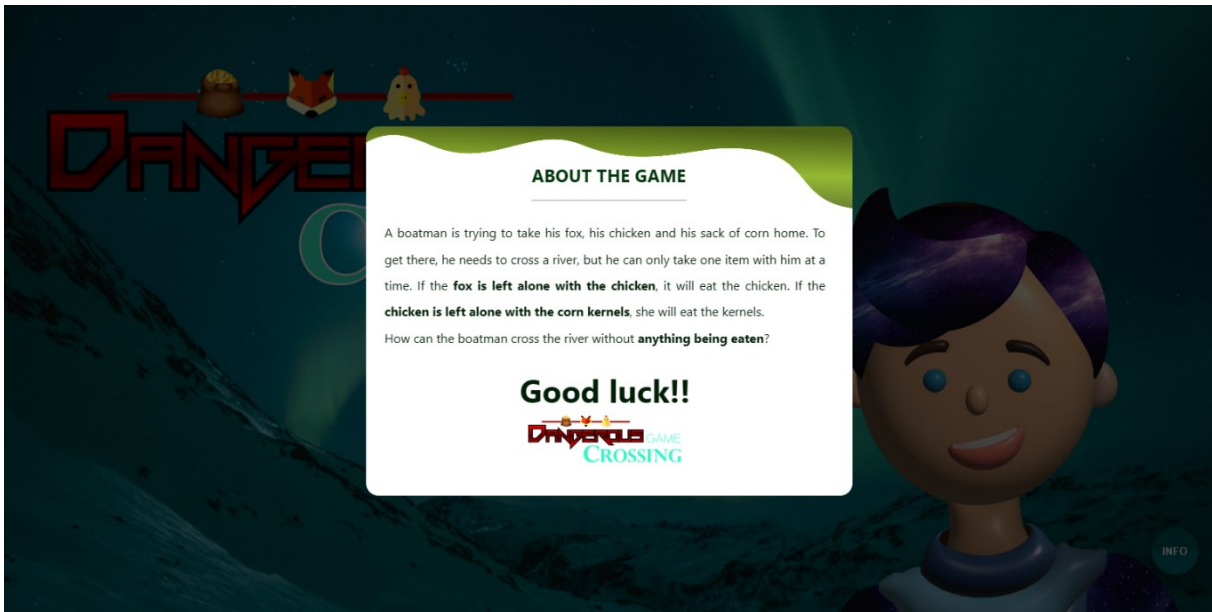
Fonte: Elaborado pelo autor

Figura 18 – Modal de Créditos



Fonte: Elaborado pelo autor

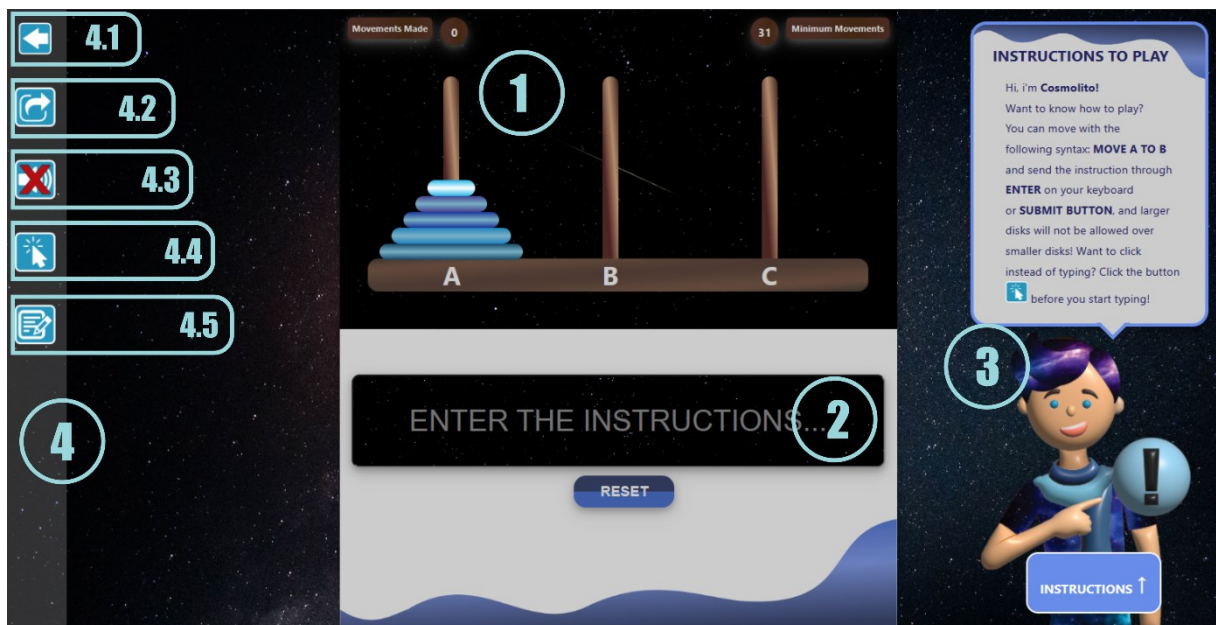
Figura 19 – Modal de Informações do Dangerous Crossing Game



Fonte: Elaborado pelo autor

Quando o aluno clica no botão “Play Game”, ele então tem acesso ao jogo como na Figura 20.

Figura 20 – Funcionamento da Tela Inicial do Tower of Hanoi Game



Fonte: Elaborado pelo autor

No marcador 1, se encontra a tela de *game*, que contém a haste com três pinos e os cinco discos do jogo, no canto superior esquerdo tem o controlador de movimentos feitos e no canto superior direito o número de movimentos mínimos. No marcador 2, tem uma caixa *input* onde o aluno irá digitar as instruções para o jogo,

fazendo assim com que ele exercite a síntese de programação e logo abaixo o botão de reset caso ele deseje retornar tudo ao estado inicial. No marcador 3, tem o mascote Cosmolito que fornece instruções do jogo assim que o aluno passa o mouse sobre o mesmo, e por último no marcador 4 tem o *menu* retrátil. No botão 4.1, retrai novamente o *menu*; o botão 4.2, retorna para a página home; o botão 4.3 reproduz a música do jogo para o usuário. Os botões 4.4 e 4.5 mudam o modo atual do jogo para o modo clique e o modo parágrafo, respectivamente.

Estes dois novos modos foram desenvolvidos para evitar o tédio após a conclusão do jogo, possibilitando diferentes formas de jogá-lo. No modo clique, Figura 21, o aluno deve clicar no disco que deseja mover e em seguida clicar no pino de destino, e seguir essa lógica até finalizar o jogo.

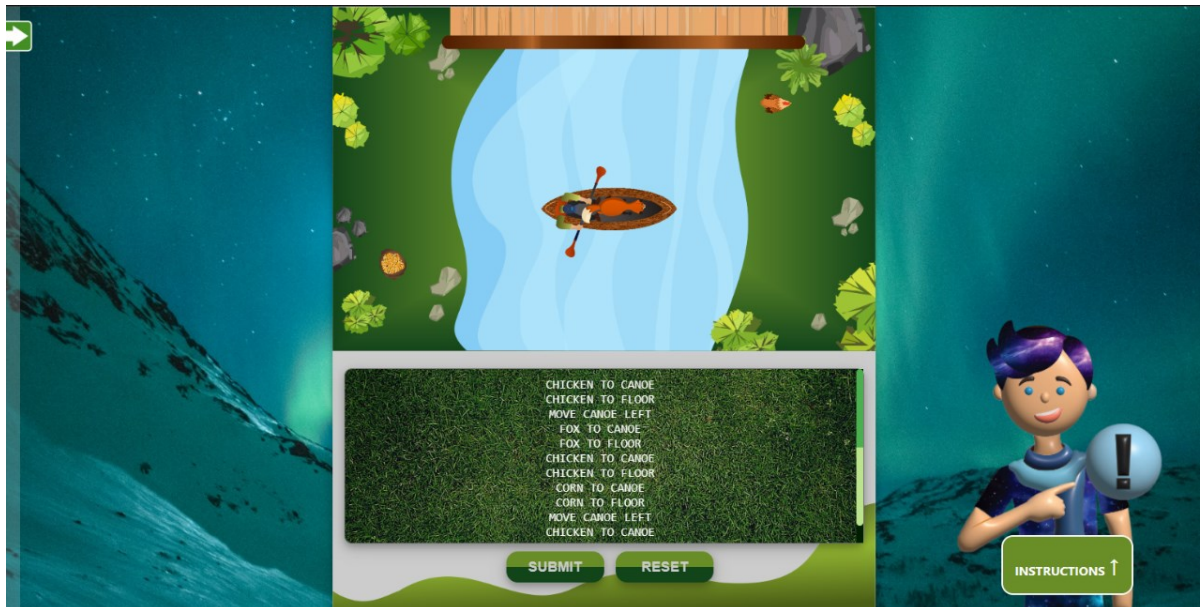
Figura 21 – Modo Clique do Tower of Hanoi Game



Fonte: Elaborado pelo autor

O modo parágrafo, por sua vez, é um modo que permite ao aluno escrever todas as instruções em sequência e, em seguida, enviá-las por meio do botão “Send” como mostra na Figura 22.

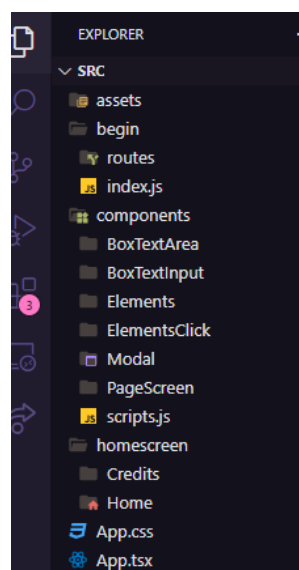
Figura 22 – Modo Parágrafo do Dangerous Crossing Game



Fonte: Elaborado pelo autor

Com o intuito de fugir dos motores mais usuais e comuns no desenvolvimento de jogos, como a *Unreal* e *Unity*, os jogos foram desenvolvidos completamente através das tecnologias Web já mencionadas neste trabalho HTML (CALDEIRA, 2015); CSS (RIBEIRO *et al.*, 2006); React (NEVES, 2023) e JavaScript/TypeScript (NASCIMENTO, 2021), e essa escolha se deu principalmente pela sua versatilidade e liberdade de execução. O projeto seguiu uma linha bem limpa e organizada para que no futuro a manutenibilidade seja fácil de execução, na Figura 23 é mostrado a estruturação dos jogos.

Figura 23 – Diretório dos jogos



Fonte: Elaborado pelo autor

O projeto está organizado em quatro pastas especificamente, sendo elas *assets*; *begin*; *components* e *homescreen*. Na pasta *assets* se encontram todos os elementos gráficos e visuais dos jogos, além dos sons que os jogos apresentam.

Na pasta *begin* é controlado as rotas realizadas pelos jogos, ou seja, os caminhos de uma página para a outra de forma sucinta.

Como o objetivo principal se trata da criação de um *framework*, o uso de componetização por meio do React é um dos pontos fundamentais para esse processo, visto a possibilidade de reutilização dos elementos em vários jogos que futuramente serão desenvolvidos.

Na pasta *components* estão todos os componentes que fazem parte dos jogos, são eles que padronizam os mesmos e são eles que permanecerão imutáveis no desenvolvimento de novos jogos. Os componentes em sua maioria seguem um padrão único e simples para facilitar nas futuras manutenções, como mostra a Figura 24.

Figura 24 – Componente de Input

```
render() {
  return (
    <div className="box-text" >
      <div className="box-counter">
        <div className="box-counter-mov">
          <div className="box-title">Movements Made</div>
          <div className="box-count">
            {this.state.contador}
          </div>
        </div>
        <div className="box-counter-min">
          <div className="box-count-min">
            31
          </div>
          <div className="box-title">Minimum Movements</div>
        </div>
      </div>
      <div className="box-text-area">
        <input className="box-text-input" id="text" placeholder="Enter the instructions..." onKeyUp={this.log} autoComplete="new-password" />
      </div>
      <div className="box-text-btns">
        <button className="box-text-btn" type="button" id="btn3" onClick={this.handleReset}>RESET</button>
      </div>
    </div>
  )
}
export default BoxText
```

Fonte: Elaborado pelo autor

4.1.2 Aplicação do *Game Design* e da Usabilidade

Atendendo a primeira das especificações de *Game Design* apresentadas anteriormente, os jogos foram desenvolvidos com base no currículo da disciplina de Algoritmos I, que visa inserir os alunos do primeiro período do curso no ambiente de

programação, por este motivo a inserção de um campo de digitação – *input* e *textarea*, conforme a Figura 25, para que o aluno consiga decompor o problema geral em partes pequenas, visto que o pensamento computacional ensina esse método.

Figura 25 – Campos de digitação dos jogos



Fonte: Elaborado pelo autor

Durante o desenvolvimento, foram consideradas as questões do *Cognitive Walkthrough* levantadas por Kieras (2006). Os jogos abordam essas questões por meio de elementos como som e movimento. Por exemplo, no Tower of Hanoi Game, quando um disco é movido, há uma animação de flutuação e, ao ser encaixado corretamente, é reproduzido um som de encaixe para proporcionar *feedback* cognitivo ao aluno.

No Dangerous Crossing Game, ao mover um item para o barco, o jogo navega automaticamente e reproduz um som de navegação. Ao retirar o item do barco e colocá-lo no chão, o jogo reproduz o som correspondente ao item, como o uivo da raposa. Esses sons e animações fornecem *feedback* ao aluno, confirmando o sucesso de suas ações.

No *design* de interface ou HUD dos jogos, os elementos foram dispostos com o objetivo de facilitar o aprendizado e agradar aos usuários, por esse motivo foi criado um mascote chamado Cosmolito para estar presente nestes e nos futuros jogos, representado na Figura 26, para aproximar os alunos do jogo e simular um "assistente virtual", proporcionando uma experiência interativa e envolvente.

As características dele foram pensadas para familiarizar os alunos com o ambiente e deixá-los mais confortáveis com a plataforma.

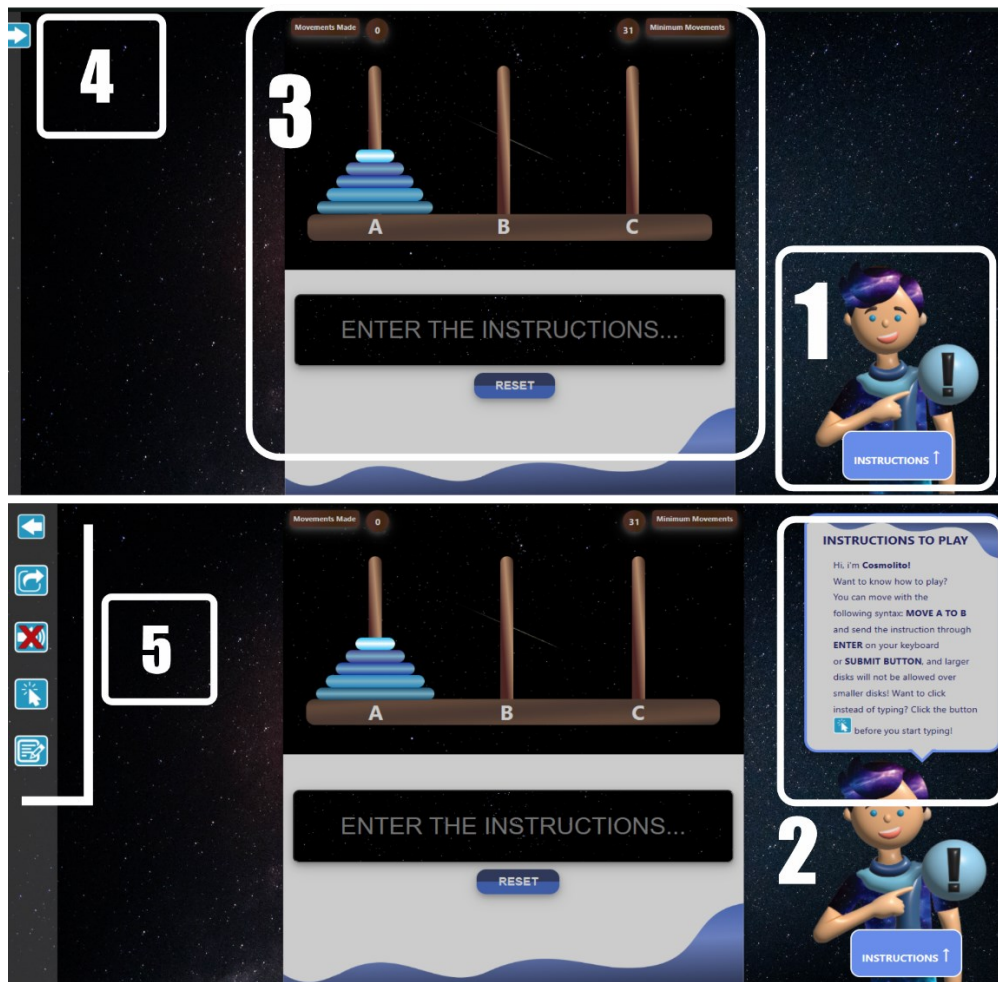
Figura 26 – Mascote dos jogos - Cosmolito



Fonte: Elaborado pelo autor

A interface foi projetada para não distrair o aluno do jogo e priorizar o aprendizado. Na Figura 27, a HUD do jogo Tower of Hanoi Game utiliza a técnica de HUD Aplicada ao Mundo do jogo, mesclada com HUD simples, por meio de seus contadores e Cosmolito em tela.

Figura 27 – HUD do Tower of Hanoi Game



Fonte: Elaborado pelo autor

O marcador 1 mostra que as instruções estão disponíveis com o mascote Cosmolito, e ao passar o *mouse* sobre o Cosmolito, as instruções são exibidas automaticamente (marcador 2), evitando a necessidade de cliques. Essa abordagem foi escolhida para evitar sobrecarregar a interface com informações excessivas e manter o foco do aluno no jogo (marcador 3). A opção de *houver* (passar o mouse) em vez de clicar foi feita para melhorar a velocidade e desempenho do jogo.

Seguindo o mesmo conceito de menos informações para não “poluir” a interface, o *menu* de opções do jogo é retrátil, marcador 4, com o objetivo de deixar os olhos do aluno voltados apenas para o jogo e, se ele desejar mudar o modo, ativar o som ambiente do jogo (já desativado) ou até mesmo retornar à página inicial, ele clica na seta e o *menu* sobressai, como mostra o marcador 5.

Quanto aos aspectos de usabilidade, para atender os fatores mencionados por Nielsen (1993), optou-se por focar nos FUFs mecanismos seguintes:

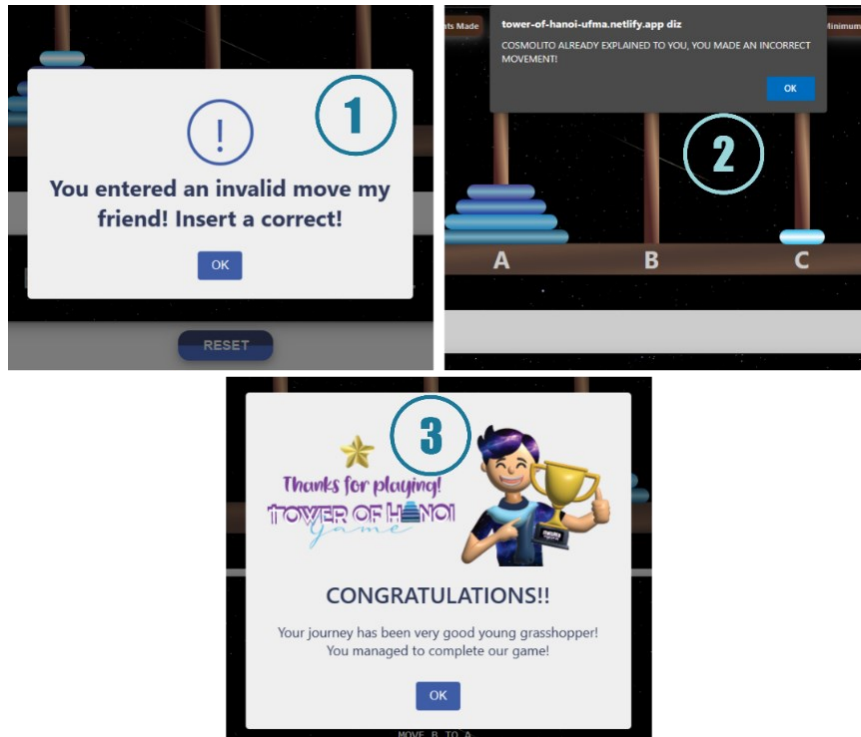
- ✓ Interação;
- ✓ Alerta;
- ✓ Abortar Operação;
- ✓ Execução passo a passo; e
- ✓ Entrada de Texto Estruturada.

Sobre a interação, que consiste em informar o usuário que o sistema registrou a sua interação, pode ser confirmado por meio do que já foi mencionando nos aspectos de sons e animações, uma vez que, o aluno sempre é informado que o jogo recebeu sua interação, comandos ou ações.

Tanto o Tower of Hanoi Game, como o Dangerous Crossing Game emitem sons após a execução do comando, mais especificamente quando a animação de encaixe dos discos termina, e quando existe o desembarque e/ou transporte dos elementos do barqueiro (raposa, galinha ou saco de milho).

Quanto ao alerta, os jogos possuem mecanismos de alerta para informar os alunos sobre ações incorretas e/ou vitórias conforme Figura 28.

Figura 28 – Mosaico com Modais de Alertas do Jogo

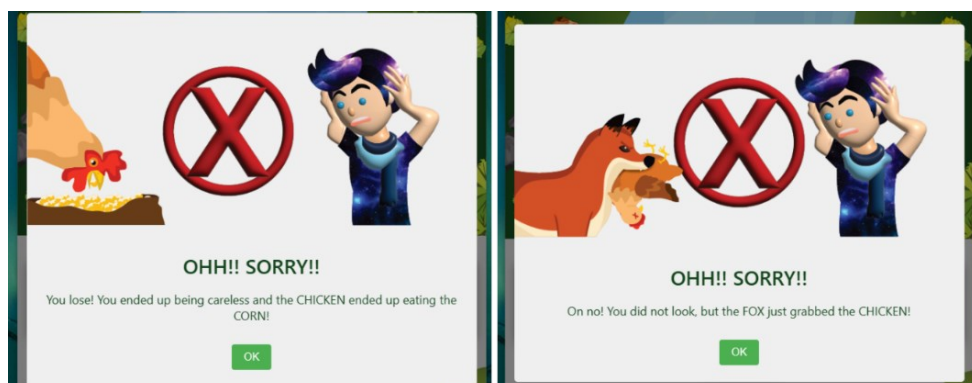


Fonte: Elaborado pelo autor

No jogo Tower of Hanoi Game, um modal é exibido automaticamente quando o aluno executa um comando inválido (marcador 1), e o jogador recebe um alerta ao tentar colocar discos maiores sobre os menores (marcador 2), além disso, ao completar o jogo movendo todos os discos para o ultimo pino seguindo as regras, o jogo apresenta um modal de parabéns (marcador 3).

O jogo Dangerous Crossing Game também segue o mesmo padrão descrito acima, contudo, ele ainda emite alertas quando o aluno movimentava a raposa ou o saco de milho em momentos incorretos resultando na “derrota” do jogador, como ilustrado na Figura 29.

Figura 29 – Mosaico com Modais de Derrotas do Dangerous Crossing Game



Fonte: Elaborado pelo autor

Para usar o mecanismo de abortar operação, que se refere a cancelar a execução de todo o sistema, foi desenvolvido um botão que permite ao usuário retornar ao ponto zero do jogo não importando onde parou, a qual foi denominado de *reset*.

Já com o objetivo de atender o mecanismo de execução passo a passo, optou-se por adicionar em cada ação executada, em ambos os jogos, um *delay* (atraso ou ação de retardar algo) para que o jogo em si consiga processar as entradas do aluno, resultando na otimização da resposta.

Por fim, quanto ao mecanismo de entrada de texto estruturada, é inserido a partir do momento que o jogo previne o aluno de cometer erros durante o *game play*, referente a entrada de dados, ou seja, o jogo informa se alguma instrução incorreta está sendo solicitada por meio dos comandos.

Com o objetivo de atender também aos alunos que possuem deficiência visual, todos os elementos gráficos de relevância possuem um *alt* (*alternative text* - descrição textual da imagem), conforme Figura 30, para que dessa forma estes possam ter uma experiência completa ao jogá-los.

Figura 30 – Representação de Alt em Imagens

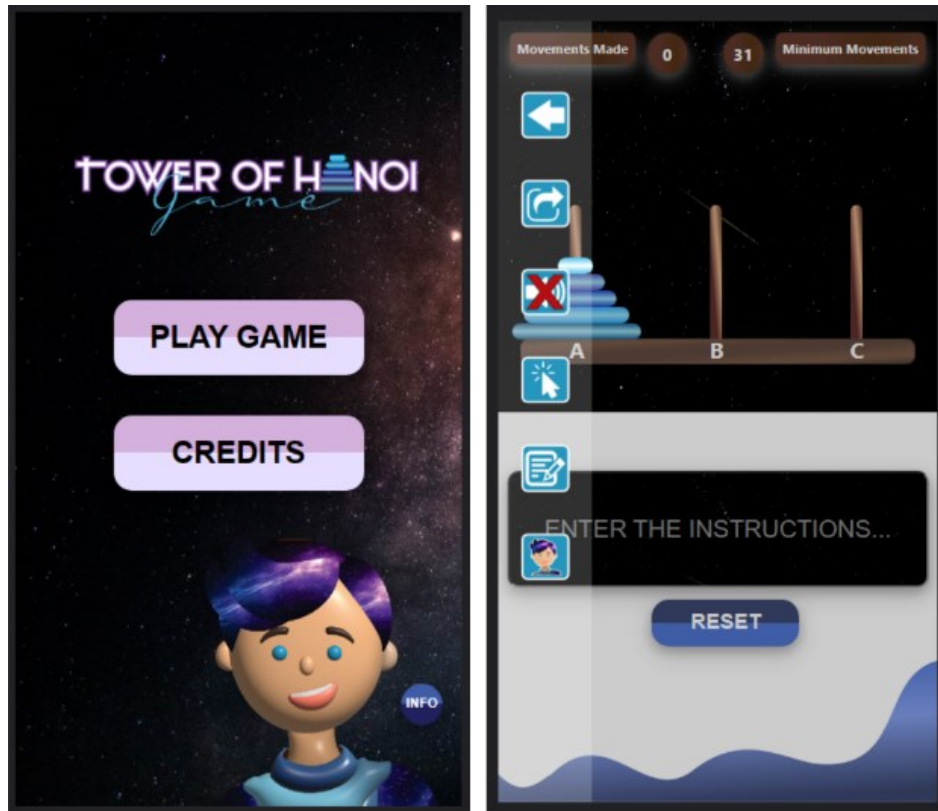
```
return (
  <StyledDiv className='content'>
    <img className='logo' src={img1} alt="Logo do game com a escrita Tower of Hanoi em caixa alta na cor branca com uma borda na cor roxa, a letra a da palavra hanoi é simbolizada por 5 discos em ordem crescente de baixo para cima, nas seguintes cores: Blumine, Boston Blue, Jacksons Purple, Victoria e Picton Blue. Um pouco abaixo, mas sobrepondo um pouco a palavra Tower of Hanoi, tem a palavra game escrita em fonte cursiva light na cor azul claro." />
    <img className='person' onMouseOver={start} onMouseOut={stop} id='voice-cosmo' src={img2} alt="Desenho do Cosmolito visto do busto para cima, um garoto aparentemente com uns 14 anos de pele clara, olhos na cor Picton Blue, sobrancelhas na cor marrom, o cabelo é dividido do lado esquerdo para o direito e cortados acima da orelha com um topete protuberante e são feitos de galaxia e a sua blusa também, por cima da mesma tem um cachecol na cor azul. O mesmo está sorrindo de olhos abertos." />
  </StyledDiv>
)
```

Fonte: Elaborado pelo autor

Além disso, os jogos são responsivos e podem ser jogados em diferentes dispositivos, incluindo *smartphones* (conforme mostrado na Figura 31), através de um navegador Web presente no dispositivo.

Mesmo com essa possibilidade, os jogos foram projetados originalmente para *desktop* e *notebook*, pois o intuito é serem aplicados em sala de aula com a monitoria do docente responsável pela disciplina a qual está sendo utilizada os jogos.

Figura 31 – Jogo Tower of Hanoi Game Versão Mobile



Fonte: Elaborado pelo autor

5 RESULTADOS

5.1.1 Diretrizes de *Game Design* e Usabilidade

As diretrizes que serão apresentadas têm como intuito nortear no desenvolvimento de jogos sérios, para que estes se tornem mais atrativos e usuais para os alunos e assim os mantenham motivados a jogar e aprender. Essas diretrizes foram projetadas tomando como base as práticas e heurísticas elencadas nas seções anteriores deste trabalho, bem como a análise após a aplicação dos jogos em sala de aula.

Abaixo, no Quadro 2, são listadas sete diretrizes que contribuirão na criação de novos jogos sérios.

Quadro 2 – Diretrizes de Game Design e Usabilidade

1. Relação com currículo/grade escolar
2. Ambiente e Elementos de aproximação do usuário
3. HUD limpa e Objetiva
4. Redução de cliques
5. Confirmação de ações executadas
6. Alertas (erros, vitórias, derrotas)
7. Acessibilidade e Responsividade*

Fonte: Elaborado pelo autor

A primeira diretriz é de crucial importância, pois o jogo tem que ser parte integrante do conteúdo ao qual ele se propõe. Ele deve ser desenvolvido levando em consideração a grade curricular da escola/faculdade onde ele será aplicado, dessa forma os objetivos ficarão bem mais claros e fáceis de serem alcançados. O jogo pode atender a apenas um assunto em específico ou até mesmo uma matéria/disciplina como um todo, e é nesse momento que estes pontos são estabelecidos.

Ao iniciar o desenvolvimento, é preciso listar e descrever aspectos que irão compor a identidade do jogo. Esses aspectos serão expostos por meio de elementos e ambientação, referente à segunda diretriz. Mesmo que os jogos sérios não tenham seu principal foco no entretenimento, é necessário que o mesmo seja atrativo e que alimente a vontade de jogar, caso contrário se tornará uma obrigação desmotivante.

A escolha de um mascote ajudará a aproximar mais os alunos ao jogo e deixá-los mais à vontade. Escolher cores que se complementam e combinam entre si, além de ser uma prática correta, é uma das formas de deixar o aluno familiarizado e não espantado com o jogo. Uma ambientação harmônica será uma extensão da imaginação do aluno e conseqüentemente o emergirá mais na game play.

A terceira diretriz é uma das mais importantes, pois ela tem ligação com o foco do aluno. A forma como serão dispostos os elementos na HUD e quais elementos deverão ficar explícitos ou subtendidos, poderá tirar ou não o foco do aluno no jogo em si. Por esse motivo, a centralização do jogo é o principal ponto, o posicionamento de botões e informações devem ficar nos cantos da tela para que a visão não seja direcionada sempre, mas sim quando necessário. Esse layout também ajudará para o alcance do mouse em caso de necessidade de clique.

Os cliques por sua vez devem ser reduzidos, como indica a quarta diretriz. Essa prática é apenas para evitar que o aluno perca tempo indo até botões e clicá-los, contudo, se o jogo necessitar de cliques para seu funcionamento não há problemas. Essa diretriz não proíbe o uso de cliques, mas sim a sua redução, principalmente em elementos além do jogo em si.

O aluno precisa ter certeza que o jogo recebeu seu comando/ação para que assim ele consiga realizar a próxima ação e prosseguir. Por isso a quinta diretriz deixa claro que é necessário que exista uma confirmação de ação executada para o aluno. Essa confirmação pode ser através de vários fatores, mas nos atentaremos aos sons e animações. O que pode parecer simples, confirmará ao aluno que sua ação foi realmente executada.

Os sons e as animações devem ser usados com cautela, para que não confunda a mente do jogador, por este motivo é crucial usá-las nos momentos de confirmação de fato, pode ser um elemento mudando de cor, girando ou até mesmo um som de "beep".

A sexta diretriz é importante para prevenir o aluno de cometer erros e se cometer que seja alertado sobre tal. Esses alertas servem para deixar os alunos atentos as suas ações e assim pensar melhor sobre seus próximos passos dentro do jogo. Deixar claro que o aluno perdeu ou ganhou é essencial para que o mesmo analise seu trajeto no jogo, podendo refazê-lo ou se dar por satisfeito com seu resultado em caso de vitória.

Por fim, a sétima e última diretriz trata-se mais de um adicional que se aplica ao desenvolvimento de quaisquer sistema, principalmente para web.

Desenvolver jogos que sejam acessíveis e responsivos englobará muito mais jogadores na experiência.

Segundo o Censo de 2010 do Instituto Brasileiro de Geografia e Estatística (IBGE), aproximadamente 46 milhões de brasileiros, cerca de 24% da população declarou possuir deficiência mental/intelectual ou tem um certo grau de dificuldade em quaisquer habilidade investigada (enxergar, ouvir, caminhar ou subir degraus).

Hoje o número de alunos nas escolas e faculdades que necessitam de recursos acessíveis para se integrarem é grande, e para os incluírem em todas as atividades é preciso aplicar elementos acessíveis, como por exemplo as descrições de imagens para os leitores de tela ou navegação por meio do teclado. E para aqueles que não têm um computador em casa, implementar o jogo voltado para o mobile também é de grande relevância para que todos consigam jogar e aprender por meio do jogo, sem se tornar 100% dependente de um computador.

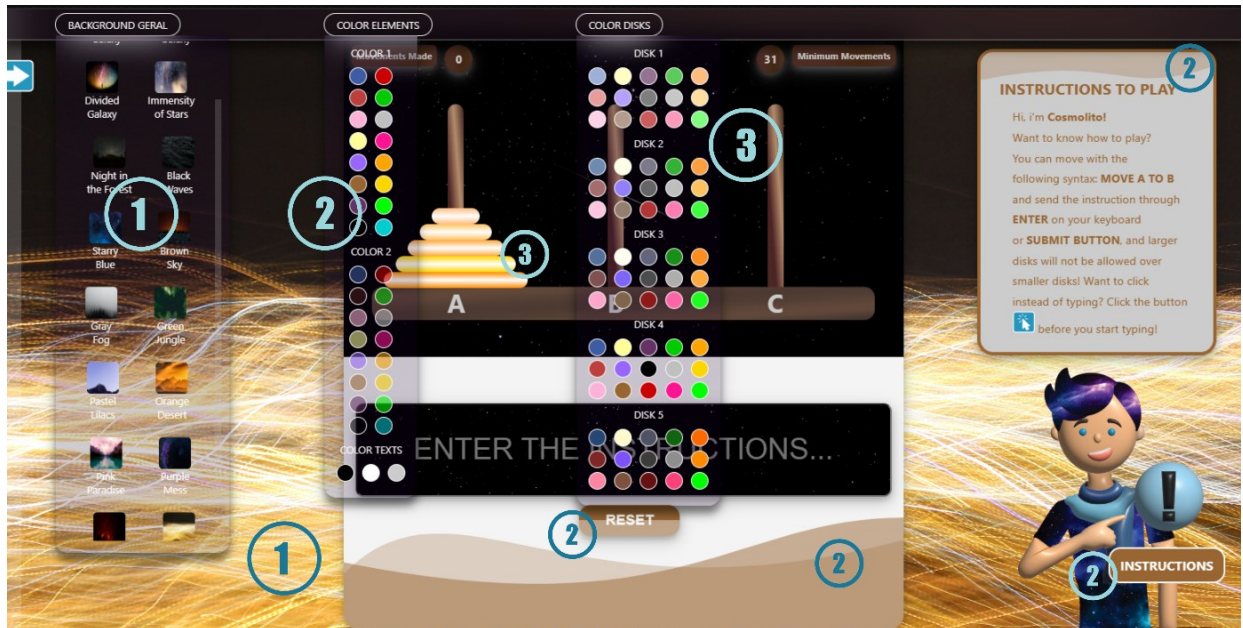
Vale ressaltar que estas diretrizes não são as únicas e muito menos tratam-se de regras, mas sim de direcionamentos que auxiliam no desenvolvimento de jogos sérios mais atrativos e usuais, estabelecidos por meio deste trabalho.

5.1.2 Framework 1.0

Por conta das semelhanças entre ambos os jogos, fortalecendo assim a possibilidade de um *framework* que os englobe, iniciou-se alterações no jogo Tower of Hanoi Game (Figura 32) para que os alunos consigam personalizá-lo conforme lhe agrade mais, podendo alterar o *background* (marcador 1), cores dos elementos e fonte (marcador 2), e as cores dos discos (marcador 3).

Vale ressaltar que ainda não houve a criação do *framework* definitivo e sim o início de alterações com intuito de uma versão beta, por este motivo o nome *framework 1.0*.

Figura 32 – Tela Game do Framework 1.0

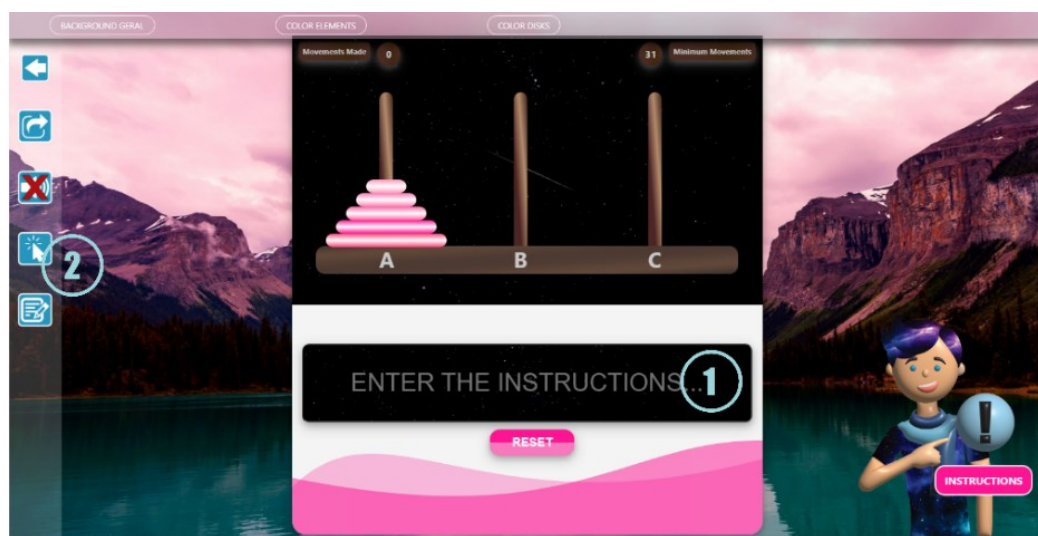


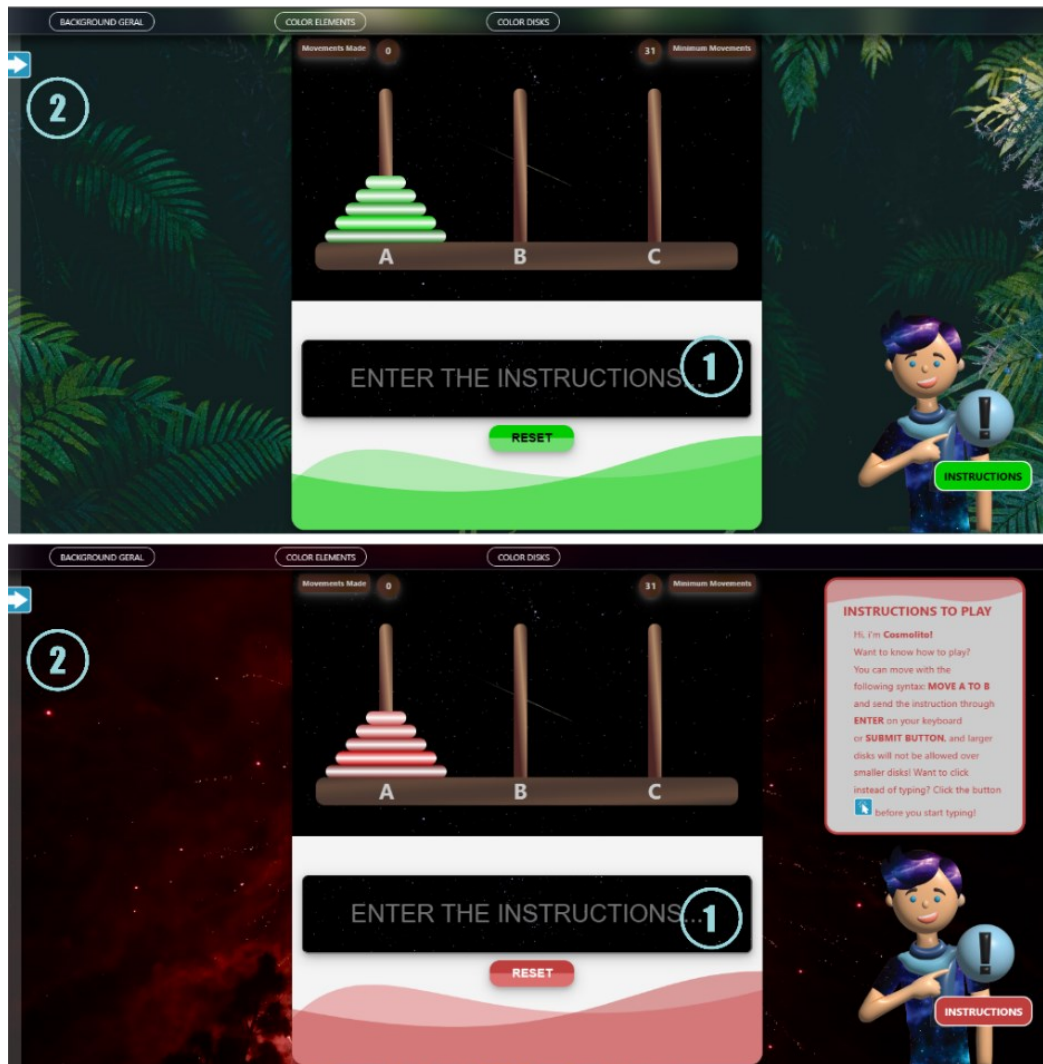
Fonte: Elaborado pelo autor

Por já existir uma padronização dos jogos por meio das diretrizes norteadoras apresentadas na Seção 5.2.1, as alterações atuais contribuem para que o aluno tenha a liberdade de atender suas necessidades pessoais e assim possivelmente ocasionar no aumento de interesse em aprender através dos jogos.

As cores e os *backgrounds* conversam entre si, como por exemplo na Figura 33, para que continue mantendo a harmonia dentro do jogo e conseqüentemente não retire o foco dos alunos no aprendizado, que é o principal ponto dos jogos desenvolvidos. Os elementos padrões dos jogos continuam sem alteração, como o *input* (marcador 1) e os elementos do *menu* (marcador 2).

Figura 33 – Variações do Framework 1.0





Fonte: Elaborado pelo autor

6 CONCLUSÃO

Neste trabalho foram investigados e apresentados diversos aspectos relacionados à jogos sérios voltados para Web, correlacionados às vantagens da criação e utilização de *frameworks*, visando a compreensão e análise aprofundada do assunto em questão. Ao longo da pesquisa, foram apresentados conceitos, diretrizes, e coleta de dados para a realização de análises críticas, o que contribuiu para o enriquecimento do conhecimento sobre o tema.

Observou-se que o uso de jogos sérios na educação tem ganho mais espaço dentro das escolas e universidades, e que não por conta deste crescimento, mas ao mesmo tempo as tecnologias Web têm emergido de forma significativa e permeado por espaços além da criação de *sites*, como é o caso do desenvolvimento de jogos Web. Essa realidade que antes não existia, agora colocou os jogos Web em uma posição que lhes permitem terem mais atenção mediante os desenvolvedores de jogos.

Devido a esse crescimento, em especial dos jogos sérios, a atenção para todas as etapas no desenvolvimento começou a ganhar mais enfoque, uma vez que existem vários elementos e diretrizes que norteiam na concepção ideal de jogos que consigam ter um resultado de aplicação bem mais eficiente. Fatores de *Game Design* e Usabilidade contribuem para a criação de jogos sérios que retirem a ideia de obrigação, e em contrapartida, contribuam para manter o aluno motivado e atraído a jogar e conseqüentemente, a aprender.

O crescimento das tecnologias Web tem contribuído cada vez mais para o surgimento de *frameworks* que utilizam das mesmas tecnologias em sua composição para a criação de outros jogos, e tudo se dá pela liberdade e versatilidade ocasionada por essas tecnologias, visto que suas exigências para funcionamento são mínimas.

No presente trabalho apresentou-se dois jogos sérios para Web denominados Tower of Hanoi Game e Dangerous Crossing Game, que foram utilizados para auxiliar no desenvolvimento do pensamento computacional de discentes da disciplina de algoritmos do primeiro período do curso de Ciência da Computação da Universidade Federal do Maranhão - UFMA. Foi-se externado o funcionamento e suas implementações, com intuito de fortalecer pontos que foram importantes para chegar no produto final. Considerando que esse trabalho conseguiu apresentar elementos e

práticas que foram utilizadas na produção dos jogos Tower of Hanoi Game e o Dangerous Crossing Game visando a criação de um *framework* para jogos sérios e ainda desenvolver diretrizes norteadoras de *Game Design* e Usabilidade para futuros desenvolvedores, os resultados somam-se positivos.

Dessa forma, é possível dizer que os objetivos deste trabalho foram alcançados, visto que, ao longo deste estudo, foi estabelecido um alicerce robusto de conhecimento que proporcionará um ponto de partida valioso para pesquisas futuras nesse campo. Ademais, obteve-se uma compreensão teórica e logo em seguida prática aprofundada acerca dos jogos sérios para Web e *frameworks*. Espera-se que este trabalho contribua para futuros desenvolvedores de jogos, em especial jogos sérios, que se atentem aos elementos do *game design* e usabilidade, usufruindo do conjunto de diretrizes apresentadas neste trabalho em seus desenvolvimentos futuros.

Como trabalhos futuros, pretende-se aprofundar-se em outros conceitos e elementos do *Game Design* e Usabilidade com intuito de englobá-los de forma coesa nas diretrizes já estabelecidas. Buscar novos e mais recentes métodos que contribuam para manter a motivação e interesse do aluno sempre presente ao ter contato com os jogos desenvolvidos, deixando a experiência enriquecedora em relação ao conhecimento pretendido e agradável ao mesmo tempo enquanto um jogo digital.

Quanto aos jogos sérios para Web pretende-se continuar aprimorando e melhorando o *framework* 1.0 já em produção, focando na criação de uma área para que o docente consiga desenvolver outros jogos no mesmo sentido, e não apenas personalizar a interface do jogo. Deixar o Cosmolito mais próximo do aluno através de uma melhor interação com mudança de expressões e possivelmente movimento. Outro ponto a ser trabalhado será o aumento das opções de personalização quanto aos *backgrounds* e as cores disponíveis, possibilitando uma variação muito mais atrativa para o aluno.

Em caso da não integração com o Cosmo, mas sim, um *framework* independente, desenvolver um sistema de cadastro e login para que o docente tenha controle dos alunos que irão utilizar, bem como do seu progresso referente ao conteúdo apresentado através do jogo, para assim se necessário realizar a aplicação novamente em sala.

É relevante enfatizar que durante a elaboração deste trabalho, a importância de ter uma atenção e estudo para desenvolver um jogo sério é algo que todos os desenvolvedores deveriam/devem ter, pois uma simples mudança em um elemento específico poderá ter um impacto significativo no objetivo pretendido com o jogo, por este motivo a necessidade de estudar e pesquisar dentro deste cenário será uma constante caminhada.

REFERÊNCIAS

ARAÚJO JÚNIOR, Mário Barbosa de. Um Estudo Comparativo de Tecnologias Web para Games. 2009. Disponível em: <<https://www.cin.ufpe.br/~tg/2009-1/mbaj.pdf>>. Acesso em: 15 de abril de 2023.

ASCHER, M. A River-Crossing Problem in Cross-Cultural Perspective. In: Mathematics Magazine, v. 63, n. 1, p. 26, 1990. DOI: 10.2307/2691506. Acesso em: 2 de maio de 2023.

BATTAIOLA, André; SANTOS, Matheus Araujo. Estudo comparativo entre modelos de game design para jogos educacionais. In: V Brazilian Symposium On Computer Games and Digital Entertainment, 2006, Recife. Digital Proceedings, Junho. Disponível em: <<https://www.cin.ufpe.br/~sbgames/proceedings/files/Analise.pdf>>. Acesso em: 1º de julho de 2023.

BIURRUM, Gabriel Régis. Desenvolvimento de Jogos Eletrônicos com HTML5. 2019. Disponível em: <<http://www2.uesb.br/computacao/wp-content/uploads/2014/09/Monografia-2.pdf>>. Acesso em: 20 de abril de 2023.

CALDEIRA, Carlos Pampulim. Introdução ao HTML (HyperText Markup Language). Universidade de Évora, 2015. Disponível em: <<https://dspace.uevora.pt/rdpc/bitstream/10174/13240/1/Introdução%20ao%20HTML.pdf>>. Acesso em: 5 de maio de 2023.

CAVALCANTE, Brunno Davisson Melo. Um framework conceitual para jogos digitais educativos solidários. Maceió, 2021. Disponível em: <<https://www.repositorio.ufal.br/bitstream/123456789/8950/1/Um%20framework%20conceitual%20para%20jogos%20digitais%20educativos%20solid%C3%A1rios.pdf>>. Acesso em: 10 de junho de 2023.

CEZAROTTO, Rangel Jungles dos; BATTAIOLA, André. Análise de Tecnologias para a Implementação de Jogos Web. In: XVI SBGames, 2017, Curitiba. SBC – Proceedings of SBGames 2017. Disponível em: <<https://www.sbgames.org/sbgames2017/papers/ArtesDesignFull/173452.pdf>>. Acesso em: 7 de julho de 2023.

CHIARATO, Ana Claudia et al. Os jogos sérios como forma de aprendizagem na saúde. In: Anais V CONEDU... Campina Grande: Realize Editora, 2018. Disponível em: <<https://editorarealize.com.br/artigo/visualizar/48480>>. Acesso em: 15 de junho de 2023.

DANCHILLA, B. Three.js Framework. In: Beginning WebGL for HTML5, 173–203. 2012. DOI: 10.1007/978-1-4302-3997-0_7.

FORTIM, Ivelise (Org). Pesquisa da Indústria Brasileira de Games 2022. ABRAGAMES. São Paulo, 2022.

GUIMARÃES, Ailamar A.; SALES, André B. de; SANTOS, Bruna A.; PALMEIRA, Eduardo G. Q. Avaliação de Características de Usabilidade em Jogos Sérios em Interação Humano-Computador. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 33., 2022, Manaus. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2022. p. 505-516. DOI: <https://doi.org/10.5753/sbie.2022.225155>.

ISO. ISO 9241-11: Requisitos Ergonômicos para Trabalho de Escritórios com Computadores - Part 11: Orientações sobre Usabilidade, 2002. Disponível em: <https://www.inf.ufsc.br/~edla.ramos/ine5624/_Walter/Normas/Parte%2011/iso9241-11F2.pdf>. Acesso em: 3 de junho de 2023.

JURISTO, Natalia; MORENO, Ana; SANCHEZ-SEGURA, Maria-Isabel. Guidelines for Eliciting Usability Functionalities. IEEE Trans. Softw. Eng. 33, 11 (November 2007), p. 744–758. DOI: doi.org/10.1109/TSE.2007.70741.

KIERAS, David. User interface design for games. Disponível em: <<https://www.cse.unr.edu/~sushil/class/games/notes/laird/User-interfaces.pdf>>. Acesso em: 2 de maio de 2023.

LIMA, Alexsandra Martins de. Torre de Hanoi e Função: a matemática pelo viés do jogo. In: Revista Tecnologia e Sociedade - 2ª Edição, 2012, Curitiba. Disponível em: <<https://periodicos.utfpr.edu.br/rts/article/viewFile/2600/1697>>. Acesso em: 5 de julho de 2023.

MARTINS, Danielle; PEREIRA, Jorge de Oliveira; NERES, Erik. Desenvolvimento de jogos 2D com Phaser e storytelling para fixação de conceitos fundamentais de programação. In: WORKSHOP DE INFORMÁTICA NA ESCOLA, 25. , 2019, Brasília. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2019. p. 471-480. DOI: <https://doi.org/10.5753/cbie.wie.2019.471>.

MORONI, Leonardo Mendes; BATTAIOLA, André Luiz. Heads-up Display integrados ao universo de um jogo eletrônico. In: SBC - Proceedings of SBGames 2011. X SBGames - Salvador - BA, November 7th - 9th, 2011. Disponível em: <<https://www.sbgames.org/sbgames2011/proceedings/sbgames/papers/art/full/92143.pdf>>. Acesso em: 1º de julho de 2023.

MOZELLI, Rodrigo. Quais são os navegadores mais usados? In: Olhar Digital. 2023. Disponível em: <<https://olhardigital.com.br/2023/05/03/internet-e-redes-sociais/quais-sao-os-navegadores-mais-usados/>>. Acesso em: 5 de junho de 2023.

NASCIMENTO, Felipe. Javascript ou Typescript?. Alura, 2021. Disponível em: <<https://site.alura.com.br/artigos/javascript-ou-typescript>>. Acesso em: 10 de maio de 2023.

NEVES, Vinícios. React: o que é, como funciona e um Guia dessa popular ferramenta JS. Alura, 2023. Disponível em: <<https://site.alura.com.br/artigos/react-js>>. Acesso em: 18 de abril de 2023.

NIELSEN, J. Usability Engineering. New York, NY: Academic Press, 1993.

OLIVEIRA, Gabriel F. de; FERREIRA, Bruna; MARQUES, Bruna A.; PALMEIRA, Anna Beatriz. USARP method: eliciting and describing USAbility Requirements with Personas and user stories. In: XXXIV Brazilian Symposium on Software Engineering, 2020, p. 497-446. DOI: <https://doi.org/10.1145/3422392.3422435>.

OLIVEIRA, Rháleff N. R. de; BELARMINO, Guilherme D.; MINHOLI, Felipe S.; RODRIGUEZ, Carla; GOYA, Denise; ROCHA, Rafaela V. Pro-AvaliaJS: Protocolo para planejamento e execução da avaliação da reação e aprendizagem de jogos sérios. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 33., 2022, Manaus. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2022. p. 517-527. DOI: <https://doi.org/10.5753/sbie.2022.225189>.

PEREIRA, André Luiz Kuczner et al. Frameworks para Desenvolvimento de Jogos: uma abordagem vantajosa no desenvolvimento de jogos eletrônicos. In: Revista Gestão em Foco, ed. 9, 2017. DOI: <https://doi.org/10.17648/unifia-gestao-foco>.

PETRI, Giani. MEEGA+: A Method for the Evaluation of the Quality of Games for Computing Education. In: SBC – Proceedings of SBGames 2019, outubro 28, 2019, Barra da Tijuca/RJ, Brasil. Disponível em: <<https://www.sbgames.org/sbgames2019/files/papers/CTDDoutorado/195720.pdf>>. Acesso em: 3 de abril de 2023.

SANTANA, Paulo Fernando Carvalho; FORTES, Denise Xavier; PORTO, Ricardo Azevedo. Jogos digitais: A utilização no processo Ensino Aprendizagem. Revista Científica da FASETE, 2016.1. Disponível em: <https://www.unirios.edu.br/revistarios/media/revistas/2016/10/jogos_digitais_a_utilizacao_no_processo_ensino_aprendizagem.pdf>. Acesso em: 20 de junho de 2023.

SANTOS, Marco Aurélio Soares dos; PEREIRA, Alice Therezinha Cybis; GONÇALVES, Marília Matos. Avaliação: análise ergonômica de interface para jogos digitais casuais para a plataforma mobile. In: XII Simpósio Brasileiro de Jogos e Entretenimento Digital, 2013. Disponível em: <<https://www.sbgames.org/sbgames2013/proceedings/>>. Acesso em: 18 de junho de 2023.

SILVA, Jamille; CRISTIANO, Fábio; MARTINS, Danielle; PEREIRA, Jorge de Oliveira; NERES, Erik. Desenvolvimento de jogos 2D com Phaser e storytelling para fixação de conceitos fundamentais de programação. In: WORKSHOP DE INFORMÁTICA NA ESCOLA, 25., 2019, Brasília. Anais [...]. Porto Alegre: Sociedade

Brasileira de Computação, 2019 . p. 471-480. DOI:
<https://doi.org/10.5753/cbie.wie.2019.471>.

TONÉIS, Cristiano Natal. Experiência estética e a interface nos jogos digitais: a produção de um edutainment game – Lua. In: Revista Tecnologia e Sociedade - 2ª Edição, 2012, Curitiba. Disponível em:
<<https://periodicos.utfpr.edu.br/rts/article/viewFile/2600/1697>>. Acesso em: 5 de junho de 2023.

TYPESCRIPT. TypeScript para o Novo Programador. 2023. Disponível em:
<<https://www.typescriptlang.org/pt/docs/handbook/typescript-from-scratch.html>>. Acesso em: 2 de abril de 2023.