

Lucas da Costa Silva

Orientador: Leonardo Pereira de Sousa

Co-orientador: Ivenilton Alexandre De Souza Moura

Gamificação do ensino. Ensinando pensamento computacional para jovens através de um jogo de quebra-cabeças

Picos - PI

23 de Setembro de 2022

Lucas da Costa Silva

Orientador: Leonardo Pereira de Sousa

Co-orientador: Ivenilton Alexandre De Souza Moura

Gamificação do ensino. Ensinando pensamento computacional para jovens através de um jogo de quebra-cabeças

Monografia apresentada ao Curso de Bacharelado em Sistemas de Informação do Campus Senador Helvídio Nunes de Barros como requisito final na aquisição do grau de Bacharel em Sistemas de Informação. Orientador: Prof. M.e Leonardo Pereira de Sousa.

Universidade Federal do Piauí
Campus Senador Heuvídio Nunes de Barros
Bacharelado em Sistemas de Informação

Picos - PI
23 de Setembro de 2022

FICHA CATALOGRÁFICA
Serviço de Processamento Técnico da Universidade Federal do Piauí
Biblioteca José Albano de Macêdo

S586g Silva, Lucas da Costa
Gamificação do ensino. Ensinando pensamento computacional para jovens através de um jogo de quebra-cabeças [recurso eletrônico] / Lucas da Costa Silva - 2022.
52 f.

1 Arquivo em PDF
Indexado no catálogo *online* da biblioteca José Albano de Macêdo-CSHNB
Aberto a pesquisadores, com restrições da Biblioteca

Trabalho de Conclusão de Curso (Graduação) – Universidade Federal do Piauí, Bacharelado em Sistemas de Informação, Picos, 2022.
“Orientador : Me. Leonardo Pereira de Sousa”
“Co-orientador : Ivenilton Alexandre de Souza Moura”

1. Programação de computadores. 2. Gamificação. 3. Jogos eletrônicos. 4. Pensamento computacional. I. Sousa, Leonardo Pereira de. II. Moura, Ivenilton Alexandre de Souza. III. Título.

CDD 005.1

Agradecimentos

Agradeço primeiramente a Deus, louvado seja, pela força, saúde, determinação e oportunidades que Ele deu a mim, graças a Ele fui capaz de encarar minhas derrotas com esperança e minhas vitórias com humildade. Rogo ao senhor que continuem protegendo e abençoando minha família, normal de seu próprio jeito. Agradeço a minha mãe, Maria das Dores, que lutou por anos para garantir seu e meu direito ao estudo e sempre (SEMPRE) esteve do meu lado, cuidando de mim, mesmo quando minhas escolhas não se alinharam as esperanças dela, agradeço a meu pai, Domingos, por sua camaradagem infinita mesmo quando longe, agradeço a meus avós, Maria Sebastiana e Antônio, meus segundos pai e mãe desde meu nascimento.

Agradeço a meus amigos, minha família escolhida por mim, Matheus, meu amigo de muitas vidas, do meu lado há décadas como amparo, Ferdinando, meu companheiro da faculdade cujos gostos nos uniram numa amizade inquebrável, Marcos Paulo, concelheiro responsável, Luiz Ricardo, Diego, Matheus Deusdará e Gabriel Ferreira, companheiros que não vejo cara a cara há muito tempo, mas que ainda vivem dentro do meu coração.

Agradeço ao Professor Ivenilton, mestre que me colocou de volta nos eixos com uma boa reprovação, Professor Ismael, que me ajudou a desenvolver minhas habilidades e me aprofundar no curso, Professor Leonardo, meu atual orientador me guiando para a conclusão do curso, e a todos os que os precederam.

Sempre teremos Ciência, Tecnologia, Engenharia e Matemática conosco. Algumas coisas sairão dos olhos do público e irão embora, mas sempre haverá ciência, engenharia e tecnologia. E sempre, sempre haverá matemática.

Katherine Johnson

Resumo

Domínio do campo conhecido como pensamento computacional é de suma importância para alunos, tanto nos cursos de informática quanto em outros cursos relacionados a exatas em geral. Enquanto há foco considerável nessa área, ela muito raramente é abordada em disciplina convencional, sendo relegada a apenas um conteúdo estudado algumas poucas vezes dentro do percurso estudantil, o que acarreta diversos problemas no aprendizado de outras disciplinas essenciais, como algoritmos que normalmente é introduzida logo no início de cursos. Logo educadores buscam maneiras de ensinar e reforçar tais conhecimentos e para tal foi decidido um método de gamificação, uma forma de ensino que faz uso de jogos especialmente produzidos com intuito de educação, que vem ganhando força graças a sua facilidade em manter interesse e atenção dos alunos, numa tentativa de mantê-los motivados durante o período introdutório do curso conhecido pelo seu alto índice de reprovações e desistências. Este trabalho propôs o desenvolvimento de um jogo eletrônico disponível em multiplataforma para ensinar, promover e reforçar práticas essenciais para o desenvolvimento de software em geral incorporadas no pensamento computacional, através de um esquema de quebra-cabeças que exige observação e planejamento.

Palavras-chaves: Ensino. Gamificação. Jogos Eletrônicos. Pensamento Computacional. Programação.

Abstract

Mastery of the field known as computational thinking is of paramount importance for students, both in computer courses and in other science-related courses in general. While there is considerable focus on this area, it is very rarely addressed in a conventional discipline, being relegated to only a content studied a few times within the student path, which causes several problems in the learning of other essential subjects, such as algorithms that are usually introduced at the start of such courses. Therefore educators search ways to teach and reinforce such knowledge and for that a gamification method was decided, a way of teaching that makes use of games specially produced with the purpose of education, which has been strengthening thanks to its ability to maintain students' interest and attention, in an attempt to keep them motivated during the introductory period of the course, known for its high rate of failure and dropouts. This work proposed the development of an electronic game available in multiplatform to teach, promote and reinforce essential practices for software development in general incorporated in computational thinking, through a puzzle scheme that requires observation and planning.

Keywords: Computational Thinking. Education. Electronic Games. Gamification. Programming.

Lista de ilustrações

Figura 1 – Ambiente de desenvolvimento 3D da Unity.	27
Figura 2 – Jogo Pillars of Eternity.	27
Figura 3 – Símbolos usados no projeto.	28
Figura 4 – Cena Title do jogo.	29
Figura 5 – Cena LevelList do jogo.	29
Figura 6 – Cena Level do jogo sem um nível carregado.	30
Figura 7 – Cena Level do jogo com um nível carregado.	31
Figura 8 – Gráfico de falhas por estágio.	36
Figura 9 – Exemplo de pergunta usando Likert.	37
Figura 10 – Questionário de usabilidade utilizado.	44
Figura 11 – Fonte: Autor.	48

Lista de tabelas

Tabela 1 – Dinâmicas de jogo e suas definições.	16
Tabela 2 – Mecânicas de jogo e suas definições.	17
Tabela 3 – Componentes de jogo e suas definições.	18
Tabela 4 – Gêneros de Jogos.	19
Tabela 5 – Comparação entre trabalhos relacionados.	25
Tabela 6 – Tabela de requisitos funcionais.	32
Tabela 7 – Tabela de requisitos funcionais.	32
Tabela 8 – Tabela de requisitos funcionais.	33
Tabela 9 – Dispositivos Testados.	34
Tabela 10 –Pontos positivos e negativos levantados.	35
Tabela 11 –Pontuação individual por questão.	37

Lista de abreviaturas e siglas

NPC	Non-Playable Character
Unity	Ferramenta para desenvolvimento mídia
Visua Studio	Ferramenta para criação e edição de código
Java Script	Linguagem de programação de script em alto nível
C#	Linguagem de programação, multiparadigma, de tipagem forte
COPS	Cooperative Online Problem Solving
OO	Orientação a Objetos
SQL	Structured Query Language
Android	Sistema operacional para celulares baseado no núcleo Linux
PC	Personal Computer
SI	Sistemas da Informação
Web	Sistema de documentos em hipermídia interligados e executados na Internet
Windows	Família de sistemas operacionais desenvolvidos pela Microsoft
Mac	Família de sistemas operacionais desenvolvidos pela Apple
Linux	Sistemas operacionais que utilizam o Kernel Linux
SO	Sistema Operacional
SUS	System Usability Scale
3D	Terceira Dimensão
UI	User Interface
APK	Android Application Pack
API	Aplication Programming Interface
RAM	Random Access Memory
JSON	JavaScript Object Notation

Lista de símbolos

% Porcentagem

Sumário

1	Introdução	12
1.1	Objetivos	13
1.1.1	Objetivos Específicos	14
1.2	Composição do Trabalho	14
2	Referencial Teórico	15
2.1	Jogos	15
2.2	Elementos de jogos	16
2.3	Gêneros de Jogos	18
2.4	Gamificação	19
2.5	<i>Unity</i>	20
2.6	<i>Visual Studio</i>	20
2.7	Pensamento Computacional	21
2.8	Algoritmos	21
2.9	<i>System Usability Scale</i>	22
3	Trabalhos Relacionados	23
4	Capítulo de Desenvolvimento	26
4.1	Ferramentas e Observações	26
4.2	Construção do Jogo	26
4.2.1	Ambiente 3D	26
4.2.2	Elementos Gráficos	27
4.2.3	Cenas	29
4.3	Níveis	30
4.4	Modelagem do Jogo	31
4.4.1	Requisitos do Sistema	31
4.4.2	Regras de Negócio	31
4.4.3	Requisitos Funcionais	32
4.4.4	Requisitos Não Funcionais	33
5	Testes e Avaliação	34
5.0.1	Avaliação de Eficiência	34
5.0.2	Teste de Campo	34
5.0.3	Avaliação de Usabilidade	36
6	Conclusão	38

6.1 Trabalhos Futuros	38
Referências	39
Apêndices	43
APÊNDICE A Questionário de Usabilidade	44
APÊNDICE B Link do Jogo	49

1 Introdução

A última década foi marcada pela expansão da tecnologia no cotidiano da população, deixando um status luxo atribuído nas décadas anteriores e se tornando uma comodidade, exigindo assim adaptação por parte de empresas fornecedoras, levando a uma demanda imensa de recursos a serem investidos na tecnologia para o desenvolvimento massivo de aparatos, aplicativos e demais ferramentas indispensáveis hoje em dia. Essa demanda requer ação por parte das empresas, que, por sua vez, necessitam cada vez mais de mais mão de obra especializada, especialmente nos últimos dois anos onde uma combinação de fatores, em especial a pandemia, levou a crescimento explosivo na demanda por profissionais qualificados, atingindo valores de mais de 300% ao redor do mundo (BERTÃO, 2021) e mais 600% no Brasil (MERCADOS... , 2021). Previsões para o futuro também são bastante otimistas, sendo possível observar crescimento de aproximadamente 125% para os próximos quatro anos (RICHARDSON, 2022), um cenário extremamente positivo para o jovem atual que nasceu e cresceu em contato direto com a tecnologia.

A maioria dos brasileiros (66,3%), Aproximadamente 140 milhões, jogam jogos eletrônicos pelo menos uma vez por semana. Entre as principais plataformas de jogo, estão *smartphone* (83%), console (48,5%) e computador (42,6%). O interesse não recai somente sobre as novas gerações, adultos também representam parte do público que faz uso de games e juntos formam uma taxa considerável da população mundial (PESQUISA... , 2019). (BUCIOR, 2015) afirma que aproximadamente de 48% dos adultos acima de 50 anos jogam games, 80% destes jogam semanalmente e 45% jogam diariamente. Esses dados atraíram a atenção de muitos pesquisadores que começaram a trabalhar com a perspectiva do potencial dos jogos para fins educacionais, evidenciando entre outros pontos a relação dos jogos com a motivação e o engajamento dos indivíduos (ALVES, 2015), essas pesquisas levaram ao uso de jogos para simular o desenvolvimento de empresas, o funcionamento de uma fábrica e, mais recentemente, o aprendizado de inteligência artificiais.

Toda a conexão entre jogos, tecnologia e computadores leva múltiplos jovens a trilhar o caminho das tecnologias da informação, seja por inspiração, determinação ou pura curiosidade, eles também são visivelmente uma maioria entre alunos do ensino médio e superior em todo o mundo, alguns começando tão cedo quanto aos dez anos (MITRA, 2010). As ambições de muitos são esmagadas cedo no período de aprendizagem, os motivos são numerosos, mas o que mais se destaca é a matéria de Algoritmos, uma disciplina tão fundamental que pode ser encontrada nas grades de diversos cursos nas áreas exatas, pois um algoritmo caracteriza-se como cadeia de passos fixos e ordenados para realização de alguma tarefa (SOUSA; JÚNIOR; FORMIGA, 2014). Enquanto teoricamente muitos são capazes compreender as necessidades de algoritmos, desenvolver as capacidades de interpretação, raciocínio e criatividade é um processo complexo e exaustivo, logo por este

conhecimento em prática efetivamente é um problema generalizado, não apenas nos cursos de programação, mas em todos os cursos onde a disciplina de algoritmos é ministrada (BARCELOS; TAROUÇO; BERCHT, 2009). Diversas pesquisas apontam problemas que vão desde a dificuldade dos alunos em compreender os conceitos de programação até a falta de motivação dos mesmos em realizar adequadamente a atividade de programação (SOUZA; BATISTA; BARBOSA, 2016), acarretando um estado crítico onde o índice nacional de desistências e reprovações está perigosamente próximo de 40% nos períodos iniciais (GARCIA; LARA; ANTUNES, 2020).

A gamificação é uma técnica que surgiu das pesquisas e interesse de estudiosos, ela baseia-se no uso de jogos eletrônicos para incentivar o aprendizado de um determinado conceito a um certo público alvo (HAMARI, 2007), afinal a ludicidade é a ferramenta ideal para atrair a atenção no momento do aprendizado, trazendo brincadeiras e jogos educativos que une a diversão ao processo educativo, facilitando assim a compreensão da criança no momento da aula e incentivando a mesma a criar laços definitivos com o conhecimento, alimentando sua mente criativa com ensinamentos divertidos e didáticos (SILVA, 2018) e, também, por que os jogos motivam de diferentes maneiras a avançar em suas etapas adquirindo recompensas à medida que os desafios são superados. Graças ao contexto vivido por muitos, a Gamificação pode ser a resposta para reacender o interesse de alunos desmotivados, direcionando-os de maneira intuitiva para um aprendizado mais suave.

A área de gamificação cresceu rapidamente nos últimos anos graças a diversos de projetos de bem-sucedidos (SWACHA, 2021), enquanto os primeiros conceitos não passavam muito do uso de certas partes de jogos já existentes para atender suas propostas, como usar o sistema de moedas de um jogo para ensinar matemática, até desenvolver jogos inteiros com o objetivo de ensinar sobre o tema desejado pela equipe, tema esse que agora pertence há uma gama enorme de possibilidades, desde jogos focados exclusivamente em matérias didáticas, até jogos que ajudam a ensinar sobre prevenção de doenças e desencorajar o uso de drogas (O'DONNELL et al., 2017).

Dentro desse cenário, jogos cujo foco é o ensino da programação, se comparados com os demais, ainda são poucos (PAPAVLASOPOULOU; GIANNAKOS; JACCHERI, 2016), mesmo os que têm o ensino programação como objetivo acabam se focando apenas na lógica de resolução dos problemas propostos ao jogador, por tanto jogos que se focam em ensinar o funcionamento de linguagens de programação, ou, pelo menos, o básico de uma, se fazem necessárias nesse meio sempre crescente que é a gamificação.

1.1 Objetivos

Este projeto tem como objetivo o desenvolvimento de um jogo eletrônico multiplataforma que ensine conceitos básicos de funcionamento de uma linguagem de programação,

tais como funções, loops e condicionais, bem como lógica algorítmica para a resolução de quebra-cabeças apresentados aos jogadores.

1.1.1 Objetivos Específicos

- Ensinar lógica algorítmica, dividindo o problema proposto em passos mais simples de fácil compreensão, para solucionar dos quebra-cabeças.
- Reforçar aplicação do pensamento computacional através da arquitetura dos níveis, disponibilizando ações e opções para superar obstáculos pré-definidos.
- Produzir interface com ênfase na IHC (Interação Humano-Computador) intuitiva para crianças na faixa etária de 8 a 12 anos, proporcionando acessibilidade a uma expansão posterior no âmbito do projeto.

1.2 Composição do Trabalho

Este trabalho é composto por um total de cinco capítulos. A Introdução previamente abordada compõe o primeiro capítulo, os quatro capítulos seguintes são respectivamente Referencial Teórico, Trabalhos Relacionados, Desenvolvimento e Conclusão.

O segundo capítulo aborda conceitos, componentes e ferramentas de suma importância para o desenvolvimento deste trabalho a fim de fornecer melhor compreensão sobre os mesmos.

O terceiro capítulo contém trabalhos de relevância considerável e temas similares, entrando em detalhes quanto as suas similaridades e diferenças, além de esclarecer o que este trabalho traz de único.

O quarto capítulo expõe como o sistema desenvolvido está organizado, suas etapas de construção e funcionamento, bem como critérios de avaliação.

O quinto e último capítulo levanta a conclusão deste trabalho, apresentando os resultados obtidos bem como melhorias que possam ser feitas ao sistema criado.

2 Referencial Teórico

2.1 Jogos

Jogos são uma construção humana que envolvem fatores sociais, econômicos e culturais. De acordo com [ELKONIN \(1998\)](#), os jogos, de maneira geral, surgiram nas sociedades como forma de iniciar o trabalho em grupo e de explicar o uso de ferramentas e artefatos para as crianças e jovens. Os jogos serviram como meio de iniciação para os jovens sobre sua própria cultura e seu meio social.

[HUIZINGA \(1993\)](#) corrobora a importância do conhecimento da história dos jogos ao dizer em sua pesquisa que, em outros tempos, os jogos e divertimentos eram os meios de a sociedade aproximar seus laços coletivos e se manter unida. Pode-se dizer que os jogos evoluíram conforme as necessidades da sociedade. ([KISHIMOTO, 1993](#)), em sua pesquisa sobre a história dos jogos recorre à antropologia e à tradição oral como responsáveis pela transmissão dos jogos às diferentes gerações:

“Considerado como parte da cultura popular, o jogo tradicional guarda a produção cultural de um povo em certo período histórico. Essa cultura não oficial, desenvolvida, sobretudo, pela oralidade, não fica cristalizada. Está sempre em transformação, incorporando criações anônimas das gerações que vão se sucedendo” ([KISHIMOTO, 1993](#), pag. 15).

Esta teoria histórica demonstra que o jogo traz um fator crucial para o conceito da gamificação; ela proporciona a aprendizagem de atividades e tarefas que o jovem vai desempenhar em sua vida. Além dessa questão social, o jogo desperta certas necessidades no ser humano, tais como o prazer e a satisfação, que são pontos importantes para o conceito de gamificação e mais ainda para o processo de aprendizagem. ([HUIZINGA, 1993](#)) define o jogo como:

“uma atividade ou ocupação voluntária, exercida dentro de certos e determinados limites de tempo e espaço, segundo regras livremente consentidas, mas absolutamente obrigatórias, dotado de um fim em si mesmo, acompanhado de um sentimento de tensão e alegria”. ([HUIZINGA, 1993](#), pag. 33) .

2.2 Elementos de jogos

WERBACH; HUNTER (2012) identificaram três tipos de elementos : dinâmicas, mecânicas e componentes, como categorias aplicáveis aos estudos e desenvolvimento da gamificação. Tais categorias são organizadas em ordem decrescente de abstração de modo que cada mecânica se liga a uma ou mais dinâmicas, e cada componente a uma ou mais mecânicas ou dinâmicas.

As dinâmicas de jogos representam o mais alto nível de abstração de elementos do jogo na Tabela 1. São os temas em torno do qual o jogo se desenvolve, assim como aspectos do quadro geral do sistema de jogos levados em consideração, mas que não fazem parte diretamente do jogo. Esses elementos mostram quais são as forças subjacentes que existem em jogos (WERBACH; HUNTER, 2012).

As dinâmicas, portanto, representam as interações entre o jogador e as mecânicas de jogo e compõem os aspectos do quadro geral de uma gamificação. Devem ser gerenciadas, mas não são explicitadas obrigatoriamente no jogo (WERBACH; HUNTER, 2012).

Tabela 1 – Dinâmicas de jogo e suas definições.

Dinâmicas	Descrição
Emoções	Jogos podem criar diferentes tipos de emoções, especialmente a da diversão (reforço emocional que mantém as pessoas jogando).
Narrativa	Estrutura que torna o jogo coerente. A narrativa não necessita explicitude, como uma história em um jogo. Também pode ser implícita, na qual toda a experiência tem um propósito em si.
Progressão	Ideia de dar aos jogadores a sensação de avançar dentro do jogo.
Relacionamentos	Refere-se à interação entre os jogadores, seja entre amigos, companheiros ou adversários.
Restrições	Refere-se à limitação da liberdade dos jogadores dentro do jogo. Tanto para controlar quanto incentivar criatividade.

Mecânicas se referem a elementos mais específicos que levam às ações também mais específicas na Tabela 2. Elas orientam as ações dos jogadores em uma direção desejada delimitando que o jogador pode ou não fazer dentro do jogo (ERIKSSON; MUSIALIK; WAGNER, 2012).

Mecânicas viabilizam o funcionamento do jogo e orientam as ações do jogador; e dependendo das mecânicas utilizadas, os jogos podem ter uma variedade ampla de estilos (ZICHERMANN; CUNNINGHAM, 2011). Vários mecanismos podem estar incluídos em uma dinâmica como, por exemplo, feedback e recompensas, os quais podem dar uma sensação de progressão no jogo. Cada mecânica é uma forma de atingir uma ou mais das dinâmicas descritas. Um evento, tal como um prêmio inesperado, pode tanto estimular o senso de diversão e curiosidade dos jogadores como ser uma forma de obter novos par-

Tabela 2 – Mecânicas de jogo e suas definições.

Mecânicas	Descrição
Aquisição de recursos	O jogador pode coletar itens a fim de atingir objetivos.
Avaliação (<i>Feedback</i>)	Permite que os jogadores vejam como estão progredindo no jogo.
Chance	Resultados incertos para ações de jogadores estimulam cuidado e geram surpresa.
Cooperação e competição	Cria-se um sentimento de vitória e derrota.
Desafios	Os objetivos que o jogo define para o jogador.
Recompensas	Benefícios ganhos pelo jogador diante de uma conquista no jogo.
Transações	A compra, venda ou troca de algo entre jogadores.
Turnos	Tempo e oportunidade para cada jogador agir.
Vitória	O “estado” que define ganhar o jogo.

ticipantes ou manter os jogadores mais experientes envolvidos (WERBACH; HUNTER, 2012).

Os componentes são aplicações específicas visualizadas e utilizadas na interface do jogo na Tabela 3. Este é o nível mais concreto dos elementos de jogos e, assim como uma mecânica se liga com uma ou mais dinâmicas, vários componentes podem fazer parte de uma mecânica.

WERBACH; HUNTER (werbach2012win) salientam que os componentes descritos acima podem assumir diversas combinações, e essa escolha deve ser feita com base no que atende mais adequadamente as demandas de um determinado contexto. Combinar as dinâmicas, mecânicas e componentes de forma que sejam efetivas para um determinado objetivo é a tarefa central de um projeto de gamificação.

DETERDING (deterding2012gamification) alerta que desenvolvedores que acreditem que “gamificar” é a simples adição de recompensas e incentivos com pontos e medalhas dentro de um sistema, estão destinados a fracassar em suas aplicações.

Tabela 3 – Componentes de jogo e suas definições.

Componentes	Descrição
Avatar	Representação visual do personagem do jogador
Bens virtuais	Itens dentro do jogo que os jogadores podem coletar e usar de forma não real que ainda tem valor para eles. Obtidos tanto por dinheiro real quanto moedas internas.
Boss	Um desafio geralmente difícil no final de um nível que deve ser derrotado, a fim de avançar no jogo.
Coleções	Agrupamento de itens acumulados dentro do jogo.
Combate	Disputa que ocorre para que o jogador derrote oponentes.
Conquistas	Recompensa que o jogador recebe por fazer um conjunto de atividades específicas.
Conteúdos desbloqueáveis	A possibilidade de desbloquear e acessar certos conteúdos no jogo se pré-requisitos forem preenchidos. Jogadores precisam fazer algo específico para desbloqueá-los.
Gráfico Social	Capacidade de ver amigos que também estão no jogo e ser poder interagir com eles. Um gráfico social torna o jogo uma extensão de sua experiência de rede social.
Missão	Similar a “conquistas”. É uma noção de jogo de que o jogador deve fazer executar algumas atividades que são especificamente definidas dentro da estrutura do jogo
Níveis	Representação numérica da evolução do jogador. O nível do jogador aumenta à medida que o jogador se torna melhor no jogo.
Pontos	Ações no jogo que atribuem pontos. São muitas vezes ligadas a níveis
Presentes	A possibilidade distribuir ao jogador coisas como itens ou moeda virtual para outros jogadores
Ranking	Lista jogadores que apresentam as maiores pontuações/conquistas/itens em um jogo.
Times	Possibilidade de jogar com outras pessoas com mesmo objetivo

2.3 Gêneros de Jogos

Gêneros são grandes grupos de conceitos em geral de interesse, baseando-se em utilizar similaridades consistentes entre diversos trabalhos a fim de identificar e classificar os mesmos em categorias de fácil reconhecimento. Assim como literatura, música e arte, jogos também possuem categorias estilísticas na forma de gêneros intrínsecos a eles e, igualmente aos outros, classificar gêneros de jogos é uma tarefa complexa, uma vez que muitos jogos possuem estilos pouco definíveis ou cobrem uma variedade de estilos tão extensa que impossibilita uma classificação precisa (JIANG; ZHENG, 2020).

Embora classificar jogos tenha se tornado uma tarefa complicada e até fútil em determinados momentos, em razão de jogos modernos constantemente desafiarem estipulações

antigas e seus moldes (CLARKE; LEE; CLARK, 2017), levando parte significativa dos jogos atuais serem grandes amalgamas de gêneros, componentes, mecânicas e dinâmicas, onde leves mudanças podem acarretar experiências divergentes. CHOI et al. (2020) define que jogos podem ser vagamente agrupados em cinco supergêneros amplos, cada um possuidor de numerosos subgêneros mais conhecidos como vistos na Tabela 4.

Tabela 4 – Gêneros de Jogos.

Supergênero	Descrição
Jogos Tradicionais	Quebra cabeças, jogos de cartas, tabuleiro e outros jogos que já existiam em outra mídia constituem jogos tradicionais.
Jogos de Simulação	Jogos baseados em esportes, pilotagem, construção simples, interação com NPCs (<i>Non-Playable Character</i> , personagem não jogável) e outras atividades reais são identificados como jogos de simulação.
Jogos de Estratégia	Jogos onde a visão e planejamento baseado em informação compõem os jogos de estratégia, geralmente são divididos em turnos, onde cada jogador tem seu tempo para agir, e tempo real, onde todos os jogadores agem simultaneamente.
Jogos de Ação	Jogos de ação são definidos por um esquema de camera responsável por conectar visão e ações dos jogadores ao ambiente do jogo. O que define os subgêneros de jogos de ação é, em geral, a perspectiva pela qual são vistos, sendo estas: primeira pessoa, terceira pessoa e plataforma.
Jogos de Fantasia	Jogos de fantasia regularmente incorporam narrativa considerável e forte exploração do mundo no qual se encontram, tendo um ritmo mais lento para que seus jogadores possam experimentar com mais atenção, quase sempre divididos na sua escala de jogadores.

2.4 Gamificação

Gamificação é uma área emergente do ensino, compreendendo a utilização dos mecanismos característicos dos jogos a fim de resolver os problemas de motivação e engajamento de dado grupo. Levando inúmeros pesquisadores a acreditar que, como introduzido anteriormente, jogos tem aptidão para possibilitar narrativas e contextos que proporcionam fácil geração e retenção de conhecimento, por tanto, se faz vantajosa compreensão e avanço nos processos para melhoria de serviços, objetos ou ambientes baseando-se no comportamento humano e na experiência que muitos obtêm em jogos (BUSARELLO, 2016).

Gamificação é, em especial, efetiva em áreas onde os conteúdos são de difícil assimilação, a final nessas áreas há uma pressão não falada entre alunos e professores, onde um tropeço ou falta de atenção inicial leva a um "efeito bola de neve", retardando seu

desenvolvimento futuro e dificultando cada vez mais a vida de quem as estuda. Logo, a gamificação entra em ação nessas áreas quando estágios são concluídos em sequência, induzindo a um ciclo de retorno positivo, elevando o engajamento e motivação, essa rotina de autoaprendizado dá liberdade a alunos falharem sem repercussões (ELSHIEKH; BUTGERIT, 2017). Interação entre alunos também é estimulada quando estudantes encontram um desafio e engajam em discussão a fim de resolvê-lo, mesmo onde ela não ocorre, alunos mais tímidos ou antissociais também se expressam com mais frequência.

Um abundante número de pesquisas apontam a eficiência da gamificação em seus objetivos principais com resultados relevantes, embora como aponta (MATALLAOUI et al., 2017), a eficácia da gamificação decresce em direta oposição a idade, sendo mais efetiva em crianças, adolescentes e jovens e menos efetiva em adultos e idosos. Parcela considerável de pesquisas estabelecem, em adicional, que a gamificação pode ser usada até na medicina, em especial na reabilitação de pessoas com distúrbios neurodivergentes e outras condições graves de deficiência, tanto dentro quanto fora de clínicas (ADLAKHA; CHHABRA; SHUKLA, 2020).

2.5 *Unity*

A *Unity* é uma plataforma de desenvolvimento da espécie *Game Engine* (Motor de Jogos), originalmente voltada para a construção de jogos na segunda e terceira dimensões, ela expandiu seu âmbito para aplicação em outras áreas como arquitetura e cinema. A *Unity* também proporciona recursos multiplataforma, ou seja, fornece suporte para o desenvolvimento de jogos para celulares, consoles, computadores e navegadores mesmo em suas licenças mais simples (ŠMÍD, 2017). Disponível tanto em planos pagos e abrangentes quanto gratuitamente para aqueles cuja renda não exceda US\$100.000 anuais (FOXMAN, 2019).

O sistema *Unity* alcançou níveis de popularidade únicos entre outras *Game Engines* graças ao próprio sistema possuir ferramentas abundantes embutidas em sua base que permitem fácil manipulação de luminosidade, som, objetos, variáveis e interface assistem intensamente no desenrolar de projetos (BUYUKSALIH et al., 2017), bem como o uso de linguagens mais simples e de fácil acesso, como *Java Script* e *C#*, para produzir a parte lógica atrás do apresentado para não desenvolvedores, a constante presença de outros desenvolvedores acolhedores e seus recursos desenvolvidos uns para os outros, e o suporte para múltiplas ferramentas e sistemas (FOXMAN, 2019).

2.6 *Visual Studio*

O *Visual Studio* é uma ferramenta de programação altamente funcional, robusta e facilmente configurável, desenvolvida tanto com foco no programador iniciante quanto o

profissional experiente, projetada para auxiliar na criação e edição de projetos *Web* e de inúmeras linguagens de programação suportadas (CHOWDHURY, 2017). O *Visual Studio* tem suporte preestabelecido com a *Unity*, o que permite uso de várias funcionalidades intrínsecas do funcionamento dentro da engine, permitindo fácil e conveniente acesso a recursos que outrora exigiriam grande esforço e uma lógica precária.

2.7 Pensamento Computacional

Pensamento computacional é o resultado obtido ao agregar pensamento crítico a fundamentos da computação com objetivo de resolver problemas, pois habilidades comumente associadas a criação e desenvolvimento de programas podem ser amplamente abstraídas para uso em inúmeras áreas (BRACKMANN, 2017). Fornecendo uma nova forma de aprender a aprender para não somente a sociedade tecnologicamente, mas para as pessoas em geral, se aproximando quase de uma linguagem escrita ou aritmética, porém usada na tentativa de representar e compreender facetas ainda mais complexas e amplas do nosso universo. WING (2021) defende que pensamento computacional pode ser considerado como uma das habilidades básicas do ser humano, no mesmo nível de andar, falar, ler e escrever, embora esteja num grau de complexidade ainda maior.

2.8 Algoritmos

Algoritmos são parte do cotidiano da população, desde receitas culinárias, instruções de consumo de medicamentos, orientações para montagem de aparelhos e moveis até processos como operações médicas constituem exemplos de algoritmos (ZIVIANI et al., 2004). É possível definir um algoritmo como sequência de ações que, quando realizadas de forma correta, levam a solução de objetivo previamente estabelecido.

Algoritmos são ferramentas importantes para resolução de problemas, como pessoas tomam planejam, decidem e exercem ações na sua rotina, desde processos simples como chegar a algum lugar ou fazer compras em um supermercado, até como organizar o orçamento de materiais para construir edifícios ou aparelhos revolucionários (CORMEN, 2017). Este processo de identificar um problema, dividi-lo em etapas de simples solução, executar essas etapas até que o grande problema seja resolvido são o núcleo da compreensão de algoritmos. Ainda segundo CORMEN (2017), seres vivos tem uma grande vantagem sobre máquinas na forma poder se adaptar a uma situação sem conhecimento prévio, onde uma máquina congelaria num imprevisto, incapaz de avançar ou retornar caso não tenha sido treinada para tal tarefa.

2.9 *System Usability Scale*

O método *System Usability Scale* (SUS) é um dos mais conhecidos métodos de averiguação do nível de usabilidade de um produto, foi desenvolvido por John Brooke em 1986 com intuito de ser usado para avaliar produtos, serviços, hardware, software, entre outros (KLUG, 2017). O SUS atingiu níveis elevados de popularidade graças a sua simplicidade, permitindo levantamento de dados de interesse para o produto sendo testado, garantindo informações cientificamente apuradas enquanto não exigindo profundidade teórica para o usuário nem para o pesquisador (GRIER et al., 2013).

O questionário de avaliação consiste de 10 perguntas, onde o usuário pode responder em uma escala de 1 a 5 no consentimento do entrevistado, onde 1 significa “Discordo Completamente” e 5 significa “Concordo Completamente” (GRIER et al., 2013), permitindo manter um escopo aproximado da satisfação dos usuários diante do produto. Após recolher as respostas, é necessário realizar uma série de cálculos matemáticos até a obter a pontuação final (KLUG, 2017):

- Para as respostas ímpares (1, 3, 5), subtrai-se 1 da pontuação obtida;
- Para as respostas pares (2 e 4), subtrai-se a resposta de 5, ou seja, numa resposta 2, pontuação de 3 deve ser contabilizada. Numa resposta 4, pontuação de 1 deve ser contabilizado;
- Os valores das individuais das dez perguntas devem ser somados e multiplicados por 2.5;
- A pontuação final pode ir de 0 a 100;

A média do System Usability Score é 68 pontos, qualquer produto que obtenha uma pontuação menor está provavelmente enfrentando problemas graves quanto a usabilidade (KLUG, 2017).

3 Trabalhos Relacionados

Nesta seção serão apresentados alguns trabalhos cuja temática e objetivos se aproximaram significativamente dos levantados neste trabalho, ou seja, auxiliar no ensino e compreensão de conceitos importantes na vida de um programador fazendo uso de estratégias de gamificação.

[SILVIANO \(2019\)](#) busca ensinar e reforçar o conhecimento sobre algoritmos através de um aplicativo desenvolvido para o sistema *Android* a fim de facilitar acesso e utilização, onde usuários contam com login e senhas para manter registro de seu progresso. O aplicativo nomeado ALGOEDUC possui um foco reservado primariamente no ensino teórico da disciplina de algoritmos, considerada como base necessária para prosseguir na programação, para então seguir com questionários de fixação.

[OLIVEIRA et al. \(2018\)](#) projetou um jogo educativo a fim de apresentar conceitos de lógica de programação e raciocínio lógico. MazeLogic, como foi chamado o projeto, trata-se de um aplicativo para celular com serviços de *login* e *ranking* e possuindo dois modos de jogo: rápido, onde o jogador é sujeito a perguntas de múltiplas escolhas divididas em módulos de dificuldade geral, e labirinto, seguindo um conceito similar, porém onde cada pergunta respondida corretamente leva o jogador mais próximo do final, sendo levado por caminhos mais longos a cada resposta incorreta. No final o desempenho do jogador é gravado e disponibilizado num ranking geral, recebendo prêmios ligados ao aprendizado, como descontos em cursos e *workshops*.

[HOLANDA; COUTINHO \(2022\)](#) produziram um jogo educativo ambicioso com o objetivo de lecionar conteúdos introdutórios nos moldes de variáveis, tipos de dados e operações, apresentados e desenvolvidos ao longo de desafios que o jogador enfrenta enquanto joga. O *World Prog* tem foco em exploração onde o jogador viaja entre planetas, cada um com um tema de conteúdo próprio, em busca de tesouros, o jogador explora cenários e enfrenta desafios lógicos baseados no planeta, sendo esses um misto entre perguntas para analisar a situação e corrigir como o problema é funciona para atingir uma resolução.

SQL Planet é o título definido para o projeto criado por [BATISTA et al. \(2019\)](#), conceituado como um jogo online de aventura de texto, isto é, um jogo clássico estruturado exclusivamente em textos escritos tanto da máquina quanto do jogador e, graças ao uso do sistema *SQLite*, permite múltiplas instâncias simultâneas. O jogo aborda um prisioneiro que deve, com uso de comandos básicos da linguagem SQL, reunir e classificar informações dos habitantes hostis e amigáveis e ambiente, cumprindo múltiplas missões de complexidade incrementante para fugir de seu planeta prisão.

O projeto elaborado por [CASAROTTO et al. \(2018\)](#) recebe o nome de Logirunner, tratando-se de um jogo multijogador em turnos, onde até quatro jogadores competem num circuito representado por um tabuleiro de xadrez de tamanho dobrado. Cada jogador,

chamado de *Runner*, começa em um ponto diferente do circuito com o objetivo atravessar os pontos de outros jogadores e retornar a seu ponto de partida, fazendo uso de cartas com funções de movimentação, condicional e repetição além de cartas que atrapalham adversários, tudo dividido em turnos com etapas onde jogadores podem, respectivamente, modificar seu código, executá-lo e fazer outras ações, como criar obstáculos.

PROSOLVE é um jogo educacional baseado na *Web*, feito por [MATHEW; MALIK; TAWAFK \(2019\)](#) com o objetivo de aperfeiçoar a habilidade de resolução de problemas de iniciantes na área de programação. Baseado numa técnica de pseudocódigo, ele cobre elementos básicos das linguagens C/C++, entrada e saída, seleção, repetição, funções e vetores. O jogo consiste de duas caixas de texto, uma com linhas de código embaralhadas e outra contendo como elas deveriam estar ou como a saída do código deveria ser, o jogador mover as linhas usando botões a fim de organizar o código, cada passo dado sendo registrado para uma pontuação final.

[BACHU; BERNARD \(2014\)](#) faz uso de estratégias de resolução cooperativa de problemas online, COPS do inglês *Cooperative Online Problem Solving*, para estimular seu público-alvo de jovens entre treze e dezessete anos no ensino de linguagens de programação. Portando tanto modo online quanto single player, mas mantendo a mesma metodologia entre ambos usando gráficos de fluxo, úteis para compreensão de códigos complexos, e pseudocódigo, para ajudar estudantes a traduzir soluções para código, transmitindo assim conceitos-chave de aritmética, operações, condicionais, laços de repetição e todos ao mesmo tempo. O jogo possui dois estilos de interação para resolver problemas propostos: Trocas, onde gráfico de fluxo e pseudocódigo se encontram desorganizados e o jogador necessita colocá-los em ordem correta, e Quebra-Cabeças, onde gráfico de fluxo e pseudocódigo se encontram espalhados e o jogador tem que encontrar partes corretas para resolver o problema.

O audacioso projeto que é Ztech de *Object-Oriented* é, nas palavras do criador [SENG; YATIM \(2014\)](#), um jogo de edutretenimento feito para guiar usuários em uma jornada relaxante de aprendizado de Programação Orientadas a Objetos. Ztech incorpora mais atenção em na área de jogo, buscando deixar esta parte mais interessante para que, em troca, o jogador esteja mais engajado quando tiver que lidar com a parte de aprendizado, pois enquanto joga ele aprenderá os conceitos essenciais de encapsulamento, herança e polimorfismo, além de exemplificar diretamente com códigos de maneira natural, graças ao fato do avatar ser um robô, entre missões onde jogadores tem seus conhecimentos testados em missões para obter recompensas.

Tabela 5 – Comparação entre trabalhos relacionados.

Trabalho Relacionado	Foco	Plataforma
(SILVIANO, 2019)	Ensino teórico	<i>Android</i>
(OLIVEIRA et al., 2018)	Ensino teórico	<i>Android</i>
(HOLANDA; COUTINHO, 2022)	Ensino teórico	Computador
(BATISTA et al., 2019)	Ensino de programação	<i>Web</i>
(CASAROTTO et al., 2018)	Pensamento Computacional	Físico
(MATHEW; MALIK; TAWAFK, 2019)	Ensino de programação	<i>Web</i>
(BACHU; BERNARD, 2014)	Ensino de programação	<i>Web</i>
(SENG; YATIM, 2014)	Ensino de programação	Computador
Este trabalho	Pensamento Computacional	<i>Android, Web</i> e Computador

Este trabalho se distingue dos citados anteriormente ao procurar ensinar, promover e reforçar práticas de importância geral para a própria área de programação na forma de: Planejamento, uma vez num estágio, o jogador deve observar o terreno, objetivos, obstáculos e ferramentas disponíveis do nível atual, da mesma forma que um programador deve se comportar diante de seus trabalhos. Algoritmização para aplicar conhecimento e código de maneira planejada da mesma forma que o jogar coloca operações. Testar se o plano e algoritmo desenvolvidos solucionam o problema e corrigir quando não.

Os trabalhos citados a cima voltaram-se em maioria para ministrar codificação, que embora seja também de suma importância, são conteúdos que exigem um domínio maior do pensamento computacional, lógica de programação e pilares teóricos, que foi o objetivo dos demais projetos salvo por um. [CASAROTTO et al. \(2018\)](#) aproximou-se bastante do proposto neste trabalho, porém sofrendo com dois problemas quando exigem presença física de tabuleiro, cartas e mais de um jogador, que embora sejam de fácil acesso, ainda precisam de planejamento e certo empenho, enquanto este trabalho pode ser acessado por qualquer um em qualquer plataforma.

4 Capítulo de Desenvolvimento

Nesta seção será apresentado o processo de desenvolvimento referente a aplicação concebida, começando na etapa de planejamento até o término da versão utilizada nos testes. Detalhando as tecnologias, métodos e funções implementadas, e como elas foram implementadas no decorrer do desenvolvimento do projeto.

4.1 Ferramentas e Observações

O jogo proposto foi desenvolvido quase inteiramente na *Unity Engine*, o que lhe garantiu acesso rápido e fácil a ferramentas de multiplataforma para publicação planejada para *Web*, *Windows*, *Mac*, *Linux* e *Android*. O aplicativo foi produzido em um *SO Windows 10 Pro* versão 21h1 baseado em 64 *bits*, enquanto a codificação foi reservada ao ambiente de desenvolvimento *Microsoft Visual Studio* versão 1.71.1.

Com sistema, ferramentas e objetivo em mãos, o jogo foi dividido em seletos níveis, cada um com cenário, objetivos, ações e opções de dificuldade e complexidade incrementante, inicialmente houve problemas com performance e robustez do jogo em máquinas mais modestas, majoritariamente devidos a maneira inicial de como os níveis eram construídos, organizados, acessados e apresentados, tanto dentro da ferramenta *Unity* quanto na aplicação independente.

Versões futuras do jogo utilizaram uma metodologia diferente para construir e apresentar níveis, abordada mais a frente, o otimizando drasticamente para que até máquinas mais modestas sejam capazes de executar o programa.

4.2 Construção do Jogo

4.2.1 Ambiente 3D

Este trabalho utiliza gráficos 3D produzidos e implementada pela própria *Unity*. O motivo por trás desta escolhas se deve a facilidade pela qual a ferramenta apresenta para produzir jogos no estilo escolhido, que envolve bastante uso de mapas com visão isométrica e uso de câmera, além de permitir modificar o apresentado até mesmo durante execução instantaneamente, como visto na Figura 1.

Devido a proposta, até mesmo os simples modelos que podem ser feitos diretamente na *Unity* permitem criatividade e conveniência enquanto um protótipo é desenvolvido. A facilidade para produzir mapas tanto com a ferramenta quanto *scripts*, bem como acesso rápido a recursos desenvolvidos tanto pelo criador quanto por outros, sejam contratados, comprados ou disponibilizados são pontos fortes.

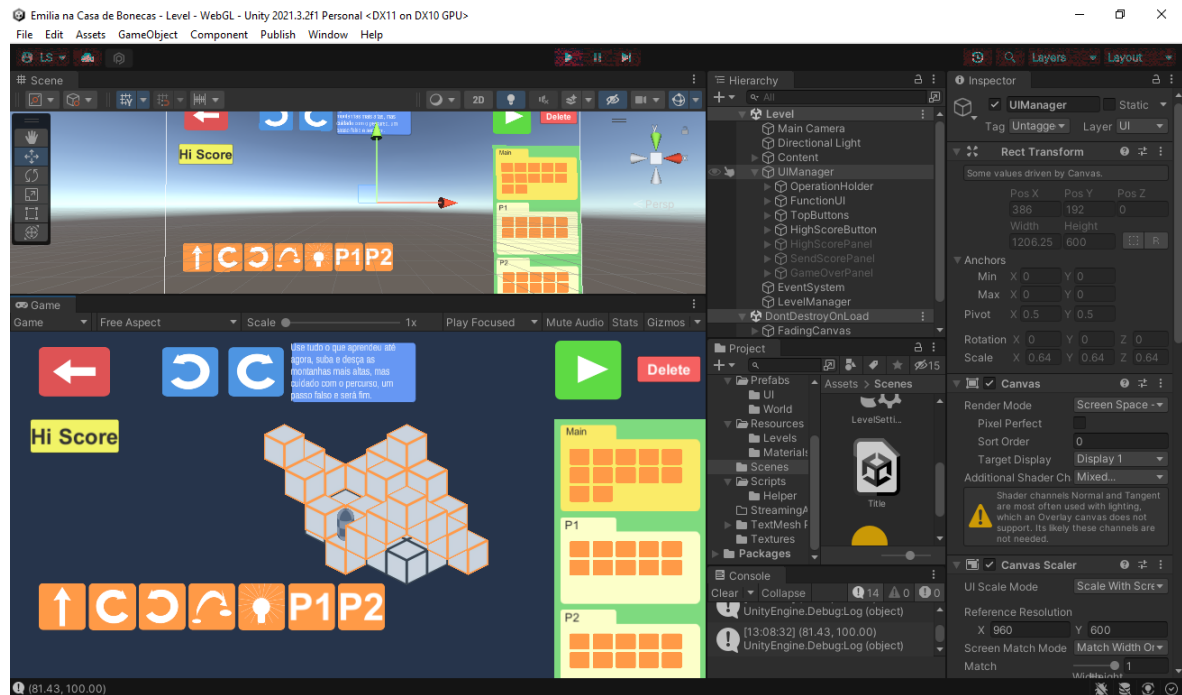


Figura 1 – Ambiente de desenvolvimento 3D da Unity.

4.2.2 Elementos Gráficos

Múltiplos símbolos são utilizados no jogo em sua totalidade, estes signos têm objetivo de melhor transmitir informações de forma intuitiva para o jogador, para isto foram usados outros jogos como inspiração para alcançar efeitos similares. Jogos como *Planescape*, *Baldur's Gate* e *Pillars of Eternity*, Figura 2, foram utilizados como referência.



Figura 2 – Jogo Pillars of Eternity.

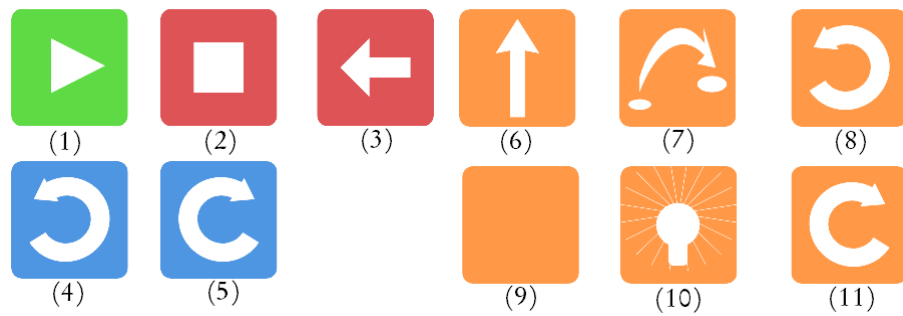


Figura 3 – Símbolos usados no projeto.

Todos os ícones implementados no jogo estão presentes na Figura 3. Onde os símbolos da esquerda representam ações gerais no jogo, enquanto os da direita representam ações que o avatar toma em um estágio.

1. É um símbolo classicamente utilizado em sites e aplicativos, comumente chamado de *play*, responsável por iniciar qualquer que seja a coisa ligada a ele, no caso o jogo deste trabalho.
2. É um símbolo comumente reconhecido como símbolo de parada, usado para pausar qualquer ação que esteja sendo tomada no jogo.
3. Representa a opção de retorno a tela anterior a presente.
4. Símbolo responsável por pôr em prática os princípios do ambiente 3D introduzidos anteriormente, sendo girar a câmera 90° no sentido horário.
5. Idêntico ao anterior, porém com a função de girar a câmera 90° no sentido anti-horário.
6. É o processo simples de andar para frente, deslocando o avatar no eixo X adjacente para o qual se encontra voltado.
7. Incorpora a ação de saltar, deslocando o avatar no eixo Y adjacente para o qual se encontra voltado.
8. Símbolo encarregado de mudar a direção a qual o avatar está virado para sua esquerda atual.
9. Tem uma função específica sendo o portador de valores textuais, utilizado tanto para escrever o nível atual quanto procedimentos e outras ações não fixas.
10. É atribuído da função de interação, quando ela é chamada em um local especial, representado por uma cor diferente, ela interage e ativa qualquer que seja o evento.
11. Símbolo encarregado de mudar a direção a qual o avatar está virado para sua direita atual.

4.2.3 Cenas

A principal maneira utilizada pela *Unity* ao dividir trabalhos é através das Cenas, grandes agrupamentos de recursos usados para estruturar menus, níveis, filmes, entre outros blocos. Cada cena é dividida e independente das demais, comportando ambiente, obstáculos, objetivos e outras ferramentas de design e construção do jogo.

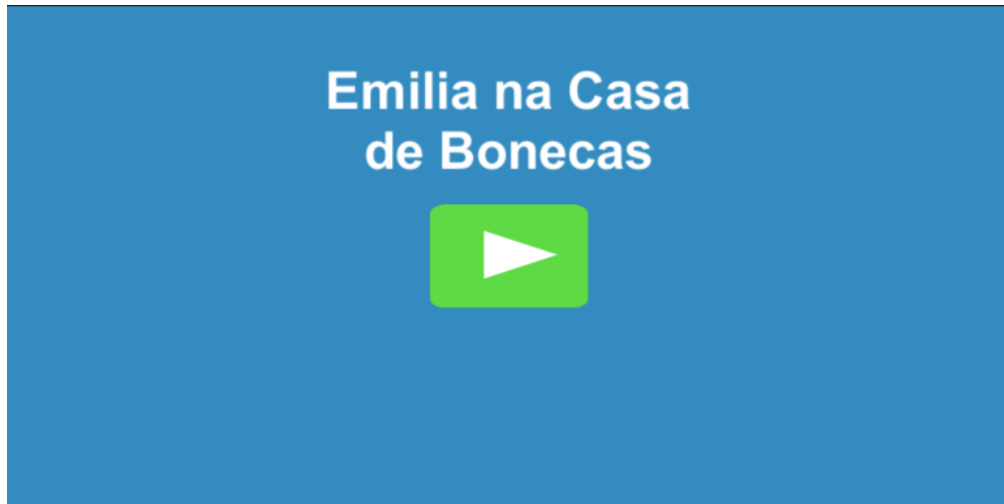


Figura 4 – Cena Title do jogo.

Há um total de três cenas utilizadas na composição deste projeto, sendo elas *Title*, representada na Figura 4, *LevelList*, representada na Figura 5, e *Level*, representada na Figura 6. *Title* age como tela de menu do jogo, onde ferramentas como configurações, informações e o botão jogar seriam retratadas, atualmente contando com o título provisório de “Emilia na Casa de Bonecas” sendo parte do enredo ainda não incorporado ao jogo. *LevelList*, sendo a tela mais simples e como o nome indica, lista os níveis disponíveis para o jogador, agindo como uma central de acesso para todos eles bem como oferecendo a opção de retorno a tela de menu.

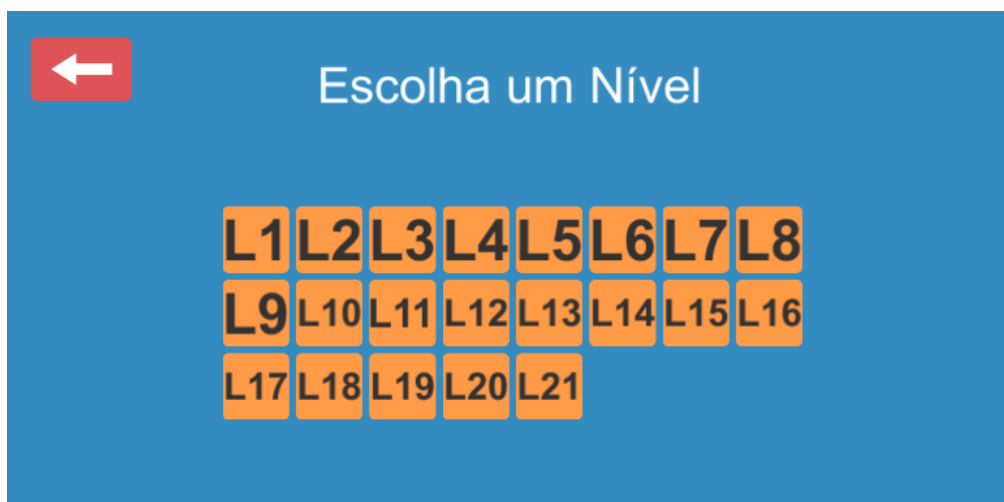


Figura 5 – Cena LevelList do jogo.

Level é a cena responsável pela representação geral dos níveis devido a metodologia implementada, porque em condições normais o esperado seria uma cena *Level* construída especificamente para cada um dos níveis, porém graças as limitações da máquina em qual o sistema fora desenvolvido, uma alternativa de otimização foi criada, e será abordada na próxima seção. A Figura 6 representa um nível básico visto pela tela de desenvolvimento da *Unity*, possuindo os componentes de interface descritos a seguir de forma visível. Além do mapa e operações disponíveis, *Level* conta com um botão de retorno para a tela de seleção de níveis no canto superior esquerdo, botões que permitem girar a câmera também se fazem presentes para fazer uso do ambiente 3D, há também um componente de texto que permitem transmitir informações de cada estágio, agindo neste caso como uma espécie de tutorial, fornecendo informações que ajudam na solução, mas não entregam respostas diretamente. No lado direito há opções de *UI* referentes à execução do jogo, executar, representado pelo botão de “play” inicia a sequência guardada no *main*, deletar remove uma opção selecionada enquanto *main* comporta as escolhas de operações, .

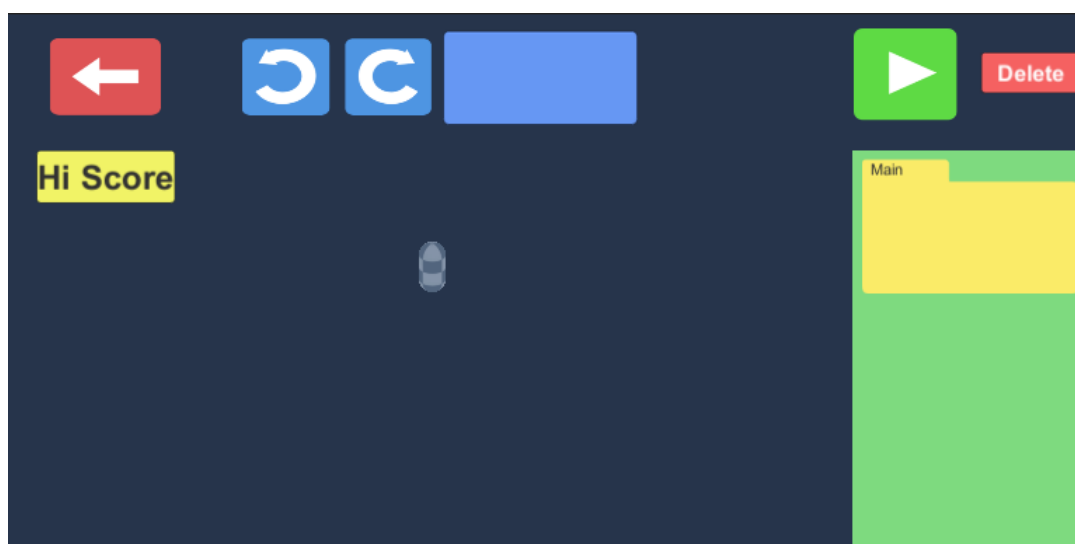


Figura 6 – Cena Level do jogo sem um nível carregado.

4.3 Níveis

Graças ao uso extensivo de códigos na *Unity* possuindo boa integração com plataformas de codificação, é possível fazer inúmeras implementações intercaladas com elementos e código, o que permitiu a estruturação de níveis modulares distintos que podem ser facilmente construídos com a relação de código e recursos. O método utilizado neste trabalho se baseia no uso de arquivos *JSON* para armazenamento de informações sobre cada um dos níveis separadamente como número de linhas e colunas, operações disponíveis, espaços na composição, número e tipo de objetivos, observações e condições de conclusão, estes arquivos são acessados pelos *scripts* ligados a *Unity*, convertendo e atribuindo valores aos

seus respectivos locais e construindo níveis uma vez acessados, isso leva a modularidade incomparável e eficiência de acesso e armazenamento. A Figura 7 apresenta um nível dentro do jogo em sua execução básica, possuindo além dos componentes previamente descritos, como é possível observar, um nível carregado por arquivo *JSON*, contendo os blocos que formam estágio, blocos amarelos que indicam objetivos, opções de ação e interação na parte inferior e o limite de ações carregados no bloco main.

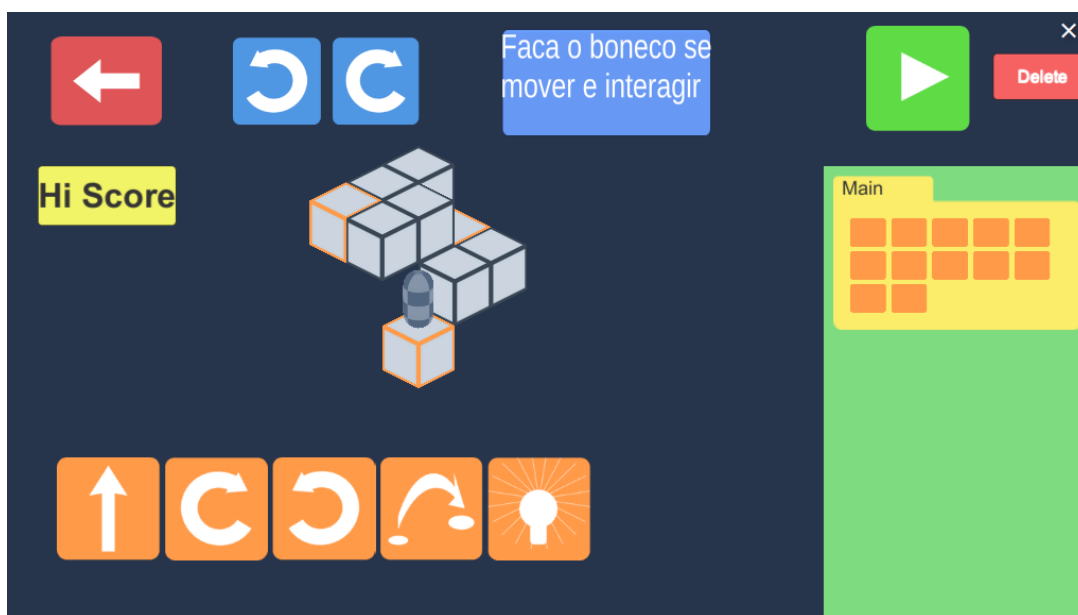


Figura 7 – Cena Level do jogo com um nível carregado.

4.4 Modelagem do Jogo

Nesta seção serão apresentados os modelos criados durante a fase de planejamento do projeto, objetivando explicar características comportamentos do jogo desenvolvido neste projeto.

4.4.1 Requisitos do Sistema

Foi realizado um processo de levantamento de requisitos a fim de determinar as principais funcionalidades a serem utilizadas no jogo onde, a partir das informações reunidas, o desenvolvimento do projeto progredisse de forma eficiente e descomplicada. Nesta seção serão descritos as regras de negócio, a seguir, os requisitos funcionais, na Tabela 7, não funcionais, Tabela 8.

4.4.2 Regras de Negócio

Regras de negócio são orientações ou limitações a cerca de como determinadas ações devem ser executadas dentro de um sistema. A Tabela 6 apresenta as principais regras de

negócio presentes neste projeto.

Tabela 6 – Tabela de requisitos funcionais.

Identificador	Descrição	Dependência
RN01	O usuário deve possuir um dispositivo compatível.	
RN02	O usuário deve possuir conexão com internet na falta de um dispositivo compatível.	RN01
RN03	O usuário deve possuir um dispositivo compatível.	

4.4.3 Requisitos Funcionais

Tabela 7 – Tabela de requisitos funcionais.

Identificador	Descrição	Dependência
RF01	O jogador deve ser capaz de iniciar o jogo.	RNF01, RNF02, RNF03, RNF04
RF02	O jogador deve ser capaz de selecionar um nível disponível.	RF01, RNF05
RF03	O jogador deve ser capaz de retornar a uma tela anterior.	RF01, RF02, RNF05
RF04	O jogador deve ser capaz de afetar o avatar dentro do jogo através das opções oferecidas.	RF02, RNF05
RF05	O jogador deve ser capaz de modificar suas ações escolhidas.	RF04, RNF05
RF06	O jogador deve ser capaz de pausar seu progresso em um nível.	RF04, RNF05
RF07	O jogador deve ser capaz de reiniciar um nível.	RF04, RF03, RNF05
RF08	O jogador deve ser capaz de abandonar um nível.	RF03, RNF05

4.4.4 Requisitos Não Funcionais

Tabela 8 – Tabela de requisitos funcionais.

Identificador	Descrição	Categoria	Dependência
RNF01	Suporte a dispositivos <i>Windows</i> a partir da versão 7, com API gráfica de DX10 ou superior.	Portabilidade	
RNF02	Suporte a dispositivos <i>Mac</i> a partir da versão High Sierra 10.13.	Portabilidade	
RNF03	Suporte a dispositivos <i>Linux</i> a partir da versão <i>Ubuntu</i> 18.04, com API gráfica de OpenGL 3.2 ou superior.	Portabilidade	
RNF04	Suporte a dispositivos <i>Android</i> a partir da versão 5.1, com API gráfica de OpenGL ES 3.0 ou superior.	Portabilidade	
RNF05	Interface simples mas informativa que permite IHC descomplicada.	Usabilidade	
RNF06	O jogador deve ser capaz de mover ângulo de câmera.	Usabilidade	RNF05
RNF07	O jogador deve ser capaz de acessar o jogo online 24 horas por dia, 7 dias por semana.	Disponibilidade	

5 Testes e Avaliação

5.0.1 Avaliação de Eficiência

O jogo foi testado em múltiplos dispositivos de variadas configurações e sistemas diferentes, buscando observar funcionamento e analisar impacto em diferentes disposições de recursos de processamento. Os dispositivos sujeitos a avaliação são detalhados na Tabela 9. O jogo foi exportado tanto na forma de *APK* quanto como arquivo executável para celulares e computadores respectivamente, ambos também foram testados na página *Web* disponibilizada para os alunos, conseqüentemente em múltiplos navegadores como: *Mozilla Firefox*, *Google Chrome*, *Ópera*, *Microsoft Edge*, *Safári* e demais navegadores populares..

Tabela 9 – Dispositivos Testados.

Modelo	Sistema Operacional	Operacional	Processador	Memória RAM
<i>Xiaomi Redmi Note 5</i>	<i>Android 10</i>		4 Core 1.8 GHz	3 GB
<i>Acer Aspire 5733</i>	<i>Windows 10 21H1</i>		i5 2.67 GHz	4 GB
Indefinido	<i>Windows 10 21H0</i>		<i>AMD Rayzen</i> 3.6 GHz	8 GB
Indefinido	<i>Linux</i> 22.02.0	<i>Ubuntu</i>	<i>AMD Rayzen</i> 3.6 GHz	8 GB
<i>Samsung Galaxy J4</i>	<i>Android 10</i>		1.4 GHz <i>Quad Core</i>	2 GB

Foi observado que o jogo funcionou corretamente em cada um dos dispositivos testados na Tabela 9, não apresentando nenhum problema visual mesmo sendo testado em diferentes esquemas e tamanhos de tela, onde a resolução máxima testada foi 16 polegadas e a mínima foi de 6 polegadas, ou problemas de execução intrínsecos do projeto em si, o projeto também está disponível para dispositivos que utilizem o *Kernel Linux*, embora não tenha sido possível obter uma máquina inteiramente dedicada para testes, eles foram realizados por meio de máquina virtual.

5.0.2 Teste de Campo

O projeto foi testado em 39 alunos do curso de Sistemas da Informação, atualmente cursando a disciplina de Algoritmos 1 sob tutela do Professor e Orientador Leonardo, um corpo demográfico diminuto consistindo de apenas uma turma, mas foi apenas o que conseguiu ser reunido. Cada aluno foi instruído a concluir os primeiros 10 níveis disponibilizados no jogo, devido ao fato de ainda estarem nas primeiras semanas do primeiro

período do curso de Sistemas da Informação, o que limitou a obtenção de dados a longo prazo devido aos limites de entrega.

Tabela 10 – Pontos positivos e negativos levantados.

Pontos positivos	Pontos negativos
Estágios com múltiplas soluções possíveis incentivam criatividade.	Limitações artísticas do projeto dificultaram compreensão.
Ambiente relaxante e agradável para experimentar.	Interface exageradamente simples, confundindo jogadores com ações similares.
Alta possibilidade de cooperação entre jogadores.	Quantidade observável de jogadores recorreram apenas obter respostas.

Primeiramente um número considerável de alunos teve problemas devido as limitações artísticas do jogo, não entendendo a orientação do avatar em alguns estágios e agindo com base nessas suposições, porém uma vez que os jogadores observaram que o avatar sempre estava voltado para a mesma direção em todos os níveis disponibilizados, algo que recebeu bastante atenção, já que todos os estágios sempre tem o avatar virado para a mesma direção, construídos especialmente para manter tudo conciso. Também foi observada certa confusão entre virar para esquerda e direita.

Entre partes que agradaram os alunos há a liberdade concebida a eles em alguns estágios, podendo usar duas ou mais resoluções diferentes em alguns casos, algo que recebeu atenção durante o desenvolvimento do jogo e continuará a receber no futuro. Outra parte foi o fato do jogo ser pouco punitivo ao permitir simplesmente reiniciar um estágio após uma falha, possibilitando um processo de tentativa e erro para resolver um estágio, especialmente em um ambiente onde os alunos não seriam penalizados de maneira alguma por errar, incentivou experimentar soluções e encarar toda a tarefa de maneira mais relaxada.

Para resolução da tarefa era esperado e foi observado cooperação entre os alunos para superar certos níveis que exigiam mais domínio, mas graças a natureza do teste ter sido realizado através da web, no site da Unity, não foi possível observar diretamente os alunos, logo não foi possível determinar quanto foi mero auxílio e quanto foi apenas entrega de respostas, mas também foi vista muita determinação e criatividade por parte de certos alunos que falharam, experimentaram e conseguiram no final usando métodos diferentes dos demais em suas soluções. A Figura 8 mostra um gráfico que representa a quantidade de falhas totais obtidas em cada estágio, onde o oitavo estágio obteve maior número falhas totais, enquanto o décimo estágio foi o qual um aluno mais falhou, num total de doze tentativas falhas.

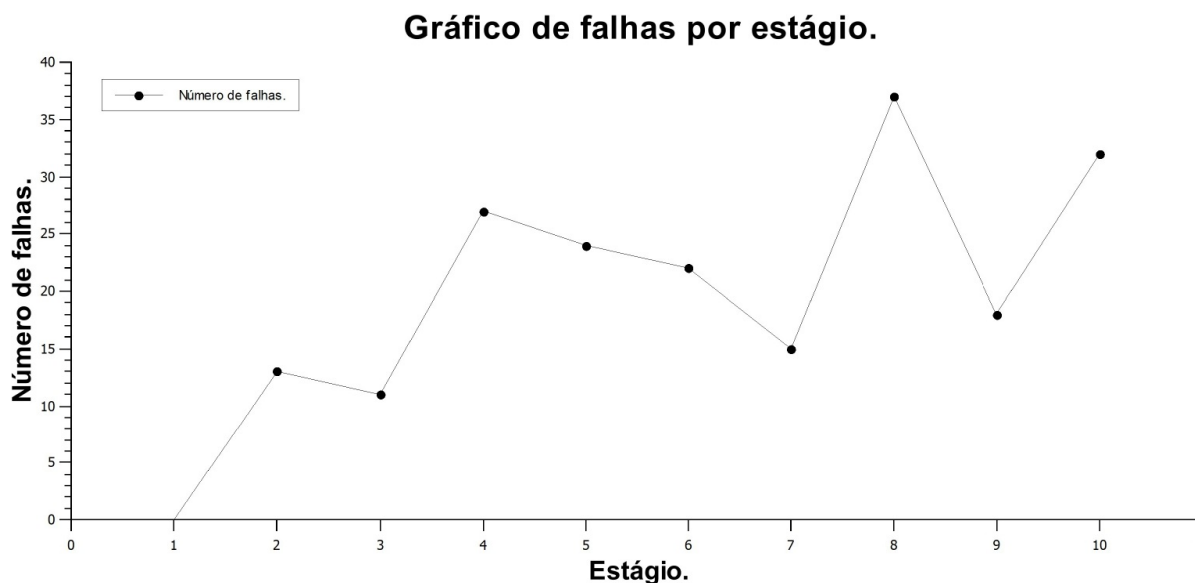


Figura 8 – Gráfico de falhas por estágio.

Infelizmente não foi possível obter dados quanto ao impacto de longo prazo e aprimoramento nas habilidades dos alunos sujeitos ao teste, em parte por eles ainda constituírem o início do primeiro período do curso de SI, em parte pela curta janela de tempo que não permitiu aprofundamento em funções e repetição, que eram esperados como próximos passos na avaliação dos alunos.

5.0.3 Avaliação de Usabilidade

Como todo e qualquer *software*, jogos são submetidos constantes testes e revisões a fim de manter nível de qualidade e usabilidade aceitáveis. Para tal é necessário manter em mente as expectativas do público-alvo e seguir suas exigências quando necessário, adaptar e aprimorar o sistema desenvolvido para acomodar as expectativas enquanto mantém uma objetividade e exatidão nas funcionalidades.

Como método de avaliação foi escolhido o método desenvolvido por Rensis Likert, [JÚNIOR; COSTA \(2014\)](#) especifica como um meio de determinar atitudes e opinião em pesquisas por meio de uma escala. O método consiste em desenvolver um conjunto de afirmações relacionadas ao tema explorado, onde os respondentes informam seu grau de concordância entre cinco níveis, um extremo portando 1 ou “Discordo Completamente” enquanto o outro 5 ou “Concordo Completamente”. [JÚNIOR; COSTA \(2014\)](#) ainda discute a presença do nível 3 ou “Indiferente/Não Concordo nem Discordo”, conhecido como ponto neutro ou de equilíbrio, é argumento que este não é um grau de concordância mensurável, sendo mais próximo de uma não resposta, porém para este trabalho foi decidida a permanência desta opção para dar mais conveniência aos respondentes.

ESTOU SATISFEITO COM O SERVIÇO RECEBIDO:				
Discordo totalmente	Discordo parcialmente	Não concordo nem discordo	Concordo parcialmente	Concordo totalmente
1	2	3	4	5

Figura 9 – Exemplo de pergunta usando Likert.

A ferramenta *Google Forms* foi escolhida para obter informações e opiniões dos usuários. O questionário disponibilizado é composto por 10 questões voltadas para usabilidade (Apêndice A), usando método aprimorado ao *SUS*, com pontuação individual representada na Tabela 11

Tabela 11 – Pontuação individual por questão.

Questão	Pontuação
1	77,5
2	78,5
3	83
4	79,5
5	86,5
6	68,5
7	80,5
8	72,5
9	75,5
10	74,5

Uma vez que a pontuação de cada questão é colhida, foi necessário calcular a média de todas as pontuações, resultando em 77,7, considerando a média do *System Usability Score* é 68 pontos para um projeto funcional. O resultado obtido no teste de usabilidade da aplicação alcançou um resultado significativamente a cima do recomendado pelo *SUS*, mostrando que a aplicação não só não atendeu os requisitos de eficiência e funcionalidade, como foi até agradável diante dos olhos dos alunos.

6 Conclusão

Este trabalho relevou o quão importante pensamento computacional é para áreas de exatas, em especial para curso de sistemas de informação, e como ferramentas que auxiliem em seu aprendizado e manutenção também são de igual relevância, a final, sua implementação faz parte das fundações necessárias para se aprofundar nessas áreas, apesar de não ter o destaque que outras disciplinas principais possuem.

Ao longo deste documento foram apresentados conceitos, metodologias e ferramentas essenciais para a construção e funcionamento do jogo. Com intuito de clarificar como ele foi desenvolvido para alcançar o proposto no início do projeto, e cumprir tais propostas com seus usuários. Para tal, testes e tarefas foram realizados com uma seleta turma, embora os resultados em campo ainda precisem de tempo e um corpo estudantil maior, consistido de mais turmas sob tutela de outros professores, para produzir dados mais concretos quanto a eficácia do projeto, é possível dizer que ele foi bem-aceito como ferramenta entre os alunos após os testes de usabilidade.

Devido a diversos problemas e limitações sofridos durante o desenvolvimento deste projeto, infelizmente não foi possível atestar com certeza a eficácia deste trabalho apenas com os dados e depoimentos obtidos nesse curto período, seria necessário realizar mais testes, por mais tempo e com mais turmas, se aprofundando em conteúdos que não puderam ser abrangidos para garantir com certeza que este trabalho pode promover mudança positiva para o aprendizado de alunos em cursos de exatas, em especial aqueles ligados a programação.

6.1 Trabalhos Futuros

Múltiplas mudanças e avanços foram idealizadas e propostas durante e após o desenvolvimento do projeto, algumas foram incorporadas, outras abandonadas e algumas postergadas, essas últimas são as abordadas nesta seção.

- Desenvolver mais níveis e aprofundar os atualmente disponíveis para comportarem mais de uma solução em maioria.
- Construir um sistema de conquistas para incentivar experimentação.
- Adicionar sistema de ranking e pontuação funcional para estimular competição e auto melhora.
- Introduzir condicionais através de mudanças controladas pelo usuário dentro do jogo.

Referências

- ADLAKHA, Shagun; CHHABRA, Deepak; SHUKLA, Pratyosh. Effectiveness of gamification for the rehabilitation of neurodegenerative disorders. *Chaos, Solitons & Fractals*, Elsevier, v. 140, p. 110192, 2020. Citado na página 20.
- ALVES, Flora. *Gamification: como criar experiências de aprendizagem engajadoras*. [S.l.]: DVS editora, 2015. Citado na página 12.
- BACHU, Eshwar; BERNARD, Margaret. Visualizing problem solving in a strategy game for teaching programming. In: CITESEER. *Proceedings of the International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS)*. [S.l.], 2014. p. 1. Citado 2 vezes nas páginas 24 e 25.
- BARCELOS, Ricardo; TAROUCO, Liane; BERCHT, Magda. O uso de mobile learning no ensino de algoritmos. *Renote*, v. 7, n. 3, p. 327–337, 2009. Citado na página 13.
- BATISTA, André Luiz França et al. Sql planet: Jogo online para ensino de linguagem sql. *SIMPÓSIO BRASILEIRO DE JOGOS E ENTRETENIMENTO DIGITAL (SBGAMES)*, p. 1220–1223, 2019. Citado 2 vezes nas páginas 23 e 25.
- BERTÃO, Naiara. *Mercado de tecnologia tem aumento de 310% de vagas em 2020*. 2021. Disponível em: <<https://valorinveste.globo.com/objetivo/empreenda-se/noticia/2021-01/10/mercado-de-tecnologia-tem-aumento-de-310percent-de-vagas-em-2020.ghtml>>. Citado na página 12.
- BRACKMANN, Christian Puhmann. Desenvolvimento do pensamento computacional através de atividades desplugadas na educação básica. 2017. Citado na página 21.
- BUCIOR, Fabio. *Agora Entertaining*. 2015. Disponível em: <<https://agoraentert.com.br/gamificacao-verdades-mitos-e-questoes-em-aberto/>>. Citado na página 12.
- BUSARELLO, Raul Inácio. *Gamification: princípios e estratégias*. [S.l.]: Pimenta Cultural, 2016. Citado na página 19.
- BUYUKSALIH, Ismail et al. 3d modelling and visualization based on the unity game engine—advantages and challenges. *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, v. 4, 2017. Citado na página 20.
- CASAROTTO, Romeu Isaac et al. Logirunner: um jogo de tabuleiro como ferramenta para o auxílio do ensino e aprendizagem de algoritmos e lógica de programação. *RENOTE*, v. 16, n. 1, 2018. Citado 2 vezes nas páginas 23 e 25.
- CHOI, Eunhye et al. Commercial video games and cognitive functions: video game genres and modulating factors of cognitive enhancement. *Behavioral and Brain Functions*, Springer, v. 16, n. 1, p. 1–14, 2020. Citado na página 19.
- CHOWDHURY, Kunal. *Mastering Visual Studio 2017*. [S.l.]: Packt Publishing Ltd, 2017. Citado na página 21.

- CLARKE, Rachel Ivy; LEE, Jin Ha; CLARK, Neils. Why video game genres fail: A classificatory analysis. *Games and Culture*, SAGE Publications Sage CA: Los Angeles, CA, v. 12, n. 5, p. 445–465, 2017. Citado na página 19.
- CORMEN, Thomas. *Desmistificando algoritmos*. [S.l.]: Elsevier Brasil, 2017. Citado na página 21.
- DETERDING, Sebastian. Gamification: designing for motivation. *interactions*, ACM, v. 19, n. 4, p. 14–17, 2012. Citado na página 17.
- ELKONIN, Daniil B. *Psicologia do jogo*. [S.l.]: Martins Fontes, 1998. Citado na página 15.
- ELSHIEKH, Rania; BUTGERIT, Laurie. Using gamification to teach students programming concepts. *Open Access Library Journal*, Scientific Research Publishing, v. 4, n. 8, p. 1–7, 2017. Citado na página 20.
- ERIKSSON, Björn; MUSIALIK, Michal; WAGNER, Justin. Gamification-engaging the future. 2012. Citado na página 16.
- FOXMAN, Maxwell. United we stand: Platforms, tools and innovation with the unity game engine. *Social Media+ Society*, SAGE Publications Sage UK: London, England, v. 5, n. 4, p. 2056305119880177, 2019. Citado na página 20.
- GARCIA, Léo Manoel Lopes da Silva; LARA, Daiany Francisca; ANTUNES, Franciano. Análise da retenção no ensino superior: um estudo de caso em um curso de sistemas de informação. *Revista da Faculdade de Educação*, v. 34, n. 2, p. 15–38, 2020. Citado na página 13.
- GRIER, Rebecca A et al. The system usability scale: Beyond standard usability testing. In: SAGE PUBLICATIONS SAGE CA: LOS ANGELES, CA. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. [S.l.], 2013. v. 57, n. 1, p. 187–191. Citado na página 22.
- HAMARI, Juho. Gamification. *The Blackwell Encyclopedia of Sociology*, Wiley Online Library, p. 1–3, 2007. Citado na página 13.
- HOLANDA, Wallace Duarte de; COUTINHO, Jarbele Cássia da Silva. World prog: Um jogo educacional para aprendizagem de conceitos básicos de programação. *RENOTE*, v. 20, n. 1, p. 213–222, 2022. Citado 2 vezes nas páginas 23 e 25.
- HUIZINGA, Johan. *Homo ludens: o jogo como elemento da cultura*. Trad. JP Monteiro. [S.l.]: São Paulo: Perspectiva, 1993. Citado na página 15.
- JIANG, Yuhang; ZHENG, Lukun. Deep learning for video game genre classification. *arXiv preprint arXiv:2011.12143*, 2020. Citado na página 18.
- JÚNIOR, Severino Domingos da Silva; COSTA, Francisco José. Mensuração e escalas de verificação: uma análise comparativa das escalas de likert e phrase completion. *PMKT–Revista Brasileira de Pesquisas de Marketing, Opinião e Mídia*, v. 15, n. 1-16, p. 61, 2014. Citado na página 36.
- KISHIMOTO, Tizuko Morchida. *Jogos infantis: o jogo, a criança e a educação*. petrópolis, rj: Ed. Vozes, 1993. Citado na página 15.

KLUG, Brandy. An overview of the system usability scale in library website and system usability testing. *Weave: Journal of Library User Experience*, Michigan Publishing, University of Michigan Library, v. 1, n. 6, 2017. Citado na página 22.

MATALLAOUI, Amir et al. How effective is “exergamification”? a systematic review on the effectiveness of gamification features in exergames. 2017. Citado na página 20.

MATHEW, Roy; MALIK, Sohail Iqbal; TAWAFK, Ragad Moufaq. Teaching problem solving skills using an educational game in a computer programming course. *Informatics in education*, Vilnius University Institute of Mathematics and Informatics, Lithuanian . . . , v. 18, n. 2, p. 359–373, 2019. Citado 2 vezes nas páginas 24 e 25.

MERCADOS de TI e dados não param de crescer e ditam profissões do futuro. 2021. Disponível em: <<https://www.cnnbrasil.com.br/business/mercados-de-ti-e-dados-nao-param-de-crescer-e-ditam-profissoes-do-futuro/>>. Citado na página 12.

MITRA, Ananda. *Digital games: Computers at play*. [S.l.]: Infobase Publishing, 2010. Citado na página 12.

O'DONNELL, Nicholas et al. How multidisciplinary is gamification research? results from a scoping review. In: *Extended Abstracts Publication of the Annual Symposium on Computer-Human Interaction in Play*. [S.l.: s.n.], 2017. p. 445–452. Citado na página 13.

OLIVEIRA, Arthur V et al. Mazelogic: Jogo educacional para ensino de lógica de programação. *Anais SULCOMP*, v. 9, 2018. Citado 2 vezes nas páginas 23 e 25.

PAPAVLASOPOULOU, Sofia; GIANNAKOS, Michail N; JACCHERI, Letizia. Creative programming experiences for teenagers: attitudes, performance and gender differences. In: *Proceedings of the The 15th International Conference on Interaction Design and Children*. [S.l.: s.n.], 2016. p. 565–570. Citado na página 13.

PESQUISA Game Brasil. 2019. Disponível em: <<https://www.pesquisagamebrasil.com.br/pesquisa-game-brasil-2019/>>. Citado na página 12.

RICHARDSON, Mary Ann. *The Fastest Growing Tech Jobs in 2022 and How to Land Them*. 2022. Disponível em: <<https://www.spiceworks.com/tech/it-careers-skills/articles/top-future-tech-jobs/>>. Citado na página 12.

SENG, Wong Yoke; YATIM, Maizatul Hayati Mohamad. Computer game as learning and teaching tool for object oriented programming in higher education institution. *Procedia-Social and Behavioral Sciences*, Elsevier, v. 123, p. 215–224, 2014. Citado 2 vezes nas páginas 24 e 25.

SILVA, MARIA DE FÁTIMA SEZARIO DA. A ludicidade como complemento da alfabetização infantil. 2018. Citado na página 13.

SILVIANO. *ALGOEDUC: um aplicativo móvel para apoio no ensino de algoritmos*. 2019. https://ufpi.br/arquivos_download/arquivos/PICOS/Not%C3%ADcias/PICOS_2022/Biblioteca/2019/Sistemas_de_Informa%C3%A7%C3%A3o_2019/M%C3%A1rcio_Silvano_de_Sousa.pdf. Accessed: 2010-09-30. Citado 2 vezes nas páginas 23 e 25.

- ŠMÍD, Antonín. Comparison of unity and unreal engine. *Czech Technical University in Prague*, p. 41–61, 2017. Citado na página 20.
- SOUSA, Bruno Jefferson; JÚNIOR, JJLD; FORMIGA, Andrei de Araújo. Introdução a programação. *João Pessoa: Editora da UFPB*, 2014. Citado na página 12.
- SOUZA, Draylson Micael; BATISTA, MH da S; BARBOSA, Ellen Francine. Problemas e dificuldades no ensino e na aprendizagem de programação: Um mapeamento sistemático. *Revista Brasileira de Informática na Educação*, v. 24, n. 1, p. 39–52, 2016. Citado na página 13.
- SWACHA, Jakub. State of research on gamification in education: A bibliometric survey. *Education Sciences*, MDPI, v. 11, n. 2, p. 69, 2021. Citado na página 13.
- WERBACH, Kevin; HUNTER, Dan. *For the win: How game thinking can revolutionize your business*. [S.l.]: Wharton Digital Press, 2012. Citado 2 vezes nas páginas 16 e 17.
- WING, Jeannette M. Pensamento computacional. *Educação e Matemática*, n. 162, p. 2–4, 2021. Citado na página 21.
- ZICHERMANN, Gabe; CUNNINGHAM, Christopher. *Gamification by design: Implementing game mechanics in web and mobile apps*. [S.l.]: "O'Reilly Media, Inc.", 2011. Citado na página 16.
- ZIVIANI, Nivio et al. *Projeto de algoritmos: com implementações em Pascal e C*. [S.l.]: Thomson Luton, 2004. Citado na página 21.

Apêndices

APÊNDICE A – Questionário de Usabilidade

Figura 10 – Questionário de usabilidade utilizado.

Questionário de Usabilidade.

Questionário referente ao jogo desenvolvido para TCC III apresentado em aula.

***Obrigatório**

1. 1. Eu acho que gostaria de jogar esse jogo com frequência em aulas. *

Marcar apenas uma oval.

- 5 Concordo Completamente
- 4 Concordo Parcialmente
- 3 Não Concordo nem Discordo
- 2 Discordo Parcialmente
- 1 Discordo Completamente

2. 2. Eu acho o jogo desnecessariamente complexo. *

Marcar apenas uma oval.

- 5 Concordo Completamente
- 4 Concordo Parcialmente
- 3 Não Concordo nem Discordo
- 2 Discordo Parcialmente
- 1 Discordo Completamente

3. 3. Eu achei o jogo fácil de usar. *

Marcar apenas uma oval.

- 5 Concordo Completamente
- 4 Concordo Parcialmente
- 3 Não Concordo nem Discordo
- 2 Discordo Parcialmente
- 1 Discordo Completamente

4. 4. Eu acho que precisaria de ajuda de uma pessoa com para jogar o jogo. *

Marcar apenas uma oval.

- 5 Concordo Completamente
- 4 Concordo Parcialmente
- 3 Não Concordo nem Discordo
- 2 Discordo Parcialmente
- 1 Discordo Completamente

5. 5. Eu acho que as interfaces e funções do jogo estão muito bem integradas. *

Marcar apenas uma oval.

- 5 Concordo Completamente
- 4 Concordo Parcialmente
- 3 Não Concordo nem Discordo
- 2 Discordo Parcialmente
- 1 Discordo Completamente

6. 6. Eu acho que o jogo apresenta muita inconsistência em seu funcionamento. *

Marcar apenas uma oval.

- 5 Concordo Completamente
- 4 Concordo Parcialmente
- 3 Não Concordo nem Discordo
- 2 Discordo Parcialmente
- 1 Discordo Completamente

7. 7. Eu imagino que as pessoas aprenderão como jogar esse jogo rapidamente. *

Marcar apenas uma oval.

- 5 Concordo Completamente
- 4 Concordo Parcialmente
- 3 Não Concordo nem Discordo
- 2 Discordo Parcialmente
- 1 Discordo Completamente

8. 8. Eu achei o jogo confuso de jogar. *

Marcar apenas uma oval.

- 5 Concordo Completamente
- 4 Concordo Parcialmente
- 3 Não Concordo nem Discordo
- 2 Discordo Parcialmente
- 1 Discordo Completamente

9. 9. Eu me senti confiante enquanto jogava o jogo. *

Marcar apenas uma oval.

- 5 Concordo Completamente
- 4 Concordo Parcialmente
- 3 Não Concordo nem Discordo
- 2 Discordo Parcialmente
- 1 Discordo Completamente

10. 10. Eu precisei aprender várias coisas novas antes de conseguir usar o sistema. *

Marcar apenas uma oval.

- 5 Concordo Completamente
- 4 Concordo Parcialmente
- 3 Não Concordo nem Discordo
- 2 Discordo Parcialmente
- 1 Discordo Completamente

APÊNDICE B – Link do Jogo

<https://play.unity.com/mg/other/webgl-builds-226780>

GAMIFICAÇÃO DO ENSINO: ENSINANDO CONCEITOS DE PROGRAMAÇÃO PARA JOVENS ATRAVÉS DE UM JOGO DE QUEBRA-CABEÇAS

LUCAS DA COSTA SILVA

Monografia _____ aprovada _____ como exigência parcial para obtenção do grau de Bacharel em Sistemas de Informação.

Data de Aprovação

Picos – PI, 07 de Outubro de 2022



Prof. Leonardo Pereira de Sousa



Prof. Ismael de Holanda Leal



Prof. Ivenilton Alexandre De Souza Moura



MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DO PIAUÍ
CAMPUS SENADOR HELVÍDIO NUNES DE BARROS
Curso de Sistemas de Informação



ATA DE APRESENTAÇÃO DE MONOGRAFIA

Período: 2022.1
Livro: 25, Folha: 02

Instala-se nesta data, 07 de outubro de 2022, a Banca Avaliadora da Monografia apresentada pelo(a) Acadêmico(a) **Lucas da Costa Silva**, sob o tema: "**Gamificação do ensino: Ensinando conceitos de programação para jovens através de um jogo de quebra-cabeças**", orientado(a) por mim **Leonardo Pereira de Sousa**, composta pelos professores **Ismael de Holanda Leal** e **Ivenilton Alexandre De Souza Moura**, presidida por mim, **Leonardo Pereira de Sousa** nos termos do Regimento Interno do Curso de Sistemas de Informação e das normas pertinentes em vigor, como cumprimento do que estabelece a Matriz Curricular e o Projeto Pedagógico do Curso de Sistemas de Informação, devidamente aprovado pela UFPI. Para o que se estabelece:

- O Concluinte disporá de no mínimo 15 (quinze) minutos e no máximo 25 (vinte e cinco) minutos para a apresentação de sua produção, para o que se solicita a não interferência;
- Os componentes da Banca, encerrada a apresentação, disporão de até 25 (vinte e cinco) minutos para a arguição, para cada componente;
- Não será permitida nenhuma manifestação da parte da plenária, antes que seja declarada encerrada a sessão.

Declaramos aberta a sessão, oportunidade em que passamos a palavra ao concluinte para sua exposição, alertando-o para o tempo já anunciado.

APÓS ANÁLISE, CONSIDERA O(A) CONCLUINTE **LUCAS DA COSTA SILVA**, COM CONCEITO/NOTA **8,0**, () APROVADO (**X**) APROVADO COM RESSALVA () REPROVADO. CASO SEJA UMA DAS DUAS PRIMEIRAS OPÇÕES O CONCLUINTE FICA CONDICIONADO AO PLENO CUMPRIMENTO DAS PROPOSIÇÕES ANUNCIADAS PELA BANCA, NO PRAZO MÁXIMO DE **10** DIAS A CONTAR DESTA DATA, COMO TEMPO MÁXIMO PARA ENTREGA DA VERSÃO FINAL NA COORDENAÇÃO DO CURSO. O NÃO CUMPRIMENTO DAS PROPOSIÇÕES DA BANCA E DO PRAZO ACIMA ESTABELECIDO IMPLICARÁ EM REPROVAÇÃO.

Nada mais havendo a registrar encerro a presente sessão, da qual lavrou-se esta ATA que vai assinada por mim, pelos componentes desta Banca e pelo(a) Concluinte.

Picos, PI, 07 de outubro de 2022.

Leonardo Pereira de Sousa
Presidente/Orientador(a)

Ismael de Holanda Leal
Membro/Avaliador(a)

Ivenilton Alexandre De Souza Moura
Membro/Avaliador(a)

Lucas da Costa Silva
Concluinte



TERMO DE AUTORIZAÇÃO PARA PUBLICAÇÃO DIGITAL NA BIBLIOTECA
"JOSÉ ALBANO DE MACEDO"

Identificação do Tipo de Documento

- () Tese
() Dissertação
 Monografia
() Artigo

Eu, Lucas da Costa Silva,
autorizo com base na Lei Federal nº 9.610 de 19 de Fevereiro de 1998 e na Lei nº 10.973 de
02 de dezembro de 2004, a biblioteca da Universidade Federal do Piauí a divulgar,
gratuitamente, sem ressarcimento de direitos autorais, o texto integral da publicação
Gamificação do Ensino. Ensinando Pensamento Computacional para Crianças
Através de um jogo de quebra-cabeças.
de minha autoria, em formato PDF, para fins de leitura e/ou impressão, pela internet a título
de divulgação da produção científica gerada pela Universidade.

Picos-PI 07 de outubro de 2022

Lucas da Costa Silva
Assinatura