

Looking for Pets: criando algoritmos para desenvolver o Pensamento Computacional

Luiz Alencar, Fernanda Pires, Marcela Pessoa

Laboratório de Pesquisa e Desenvolvimento em Tecnologias Educacionais (ThinkTED)

Escola Superior de Tecnologia – Universidade do Estado do Amazonas (EST/UEA)

Amazonas, Brazil

{lfa.lic17, fpires, mspessoa}@uea.edu.br

Abstract—This paper reports the development process of a 3D strategy game and decision making, for mobile platforms, entitled *Looking for Pets*, whose objective is to promote the development of logical reasoning, as well as to exercise the abilities of Computational Thinking through a process for the construction of algorithms. The elements and mechanics of the game are based on Constructionism. Preliminary tests point out that it can act in the pedagogical field generating positive impacts in a substantial way for learning.

Keywords-computational thinking; 3D strategy game; construction of algorithms;

Abstract—Este artigo descreve o processo de desenvolvimento de um jogo de estratégia 3D e tomada de decisão, para plataformas móveis, intitulado *Looking for Pets*, cujo objetivo é promover o desenvolvimento do raciocínio lógico e exercitar as habilidades do Pensamento Computacional através da construção de algoritmos. Os elementos e mecânica do jogo são baseados na Teoria Construcionista. Testes preliminares apontam que ele pode atuar no campo pedagógico gerando impactos substancialmente positivos para o aprendizado.

Keywords-pensamento computacional; jogos 3D; construção de algoritmos;

I. INTRODUÇÃO

A necessidade de resolver problemas acompanha a evolução da humanidade e é responsável por grandes avanços tecnológicos. A ciência busca maneiras de melhorar os processos e ajudar pessoas a se encaixar neste perfil de resolvedores de problema. Neste contexto, tem-se discutido o conceito de Pensamento Computacional que refere-se a um conjunto de habilidades que permite aos seres humanos resolver problemas de forma otimizada [1]. Essas habilidades são aplicáveis a todas as áreas do conhecimento.

O desenvolvimento do Pensamento Computacional, como habilidade de resolver problemas através de leituras críticas da realidade, é essencial na formação dos cidadãos do século XXI. Alguns países incluíram em seus currículos escolares esse componente e o Brasil está em processo de discussão de como implementar tal alteração no currículo obrigatório de formação básica, gerando discussões sobre como abordar em sala de aula ou quais abordagens pedagógicas podem ser implementadas para esse fim [2], [3].

Pesquisas recentes apresentam resultados promissores quanto à implementação de jogos para educação [4]. As características motivacionais dos jogos permitem aos indivíduos

construírem aprendizagem durante processos de resolução de problemas, em um ambiente lúdico e imersivo, delineado por regras, limites, conflitos, metas e desafios [5] em que aprender uma nova estratégia diz respeito à incorporação de um tema ao seu rol de conhecimentos.

Este artigo apresenta uma proposta de jogo educacional 3D, de gênero *puzzle*, com realidade misturada, denominado *Looking for Pets*. Seu objetivo é auxiliar no desenvolvimento do Pensamento Computacional de forma lúdica e interativa. O jogo trata da jornada de Lara que tem como missão capturar animais perdidos. Durante essa trajetória o usuário precisa visualizar os problemas e criar um algoritmo capaz de solucioná-lo. A cada fase concluída o personagem adquire pontos até alcançar seus objetivos. Sua *gameplay* tem como meta promover a aprendizagem por meio da investigação e exploração.

Este artigo está organizado da seguinte forma: na Seção II são apresentados os trabalhos relacionados; as Seções III e IV apresentam a Fundamentação Teórica associada à proposta do trabalho, tal como, conceitos relativos ao Pensamento Computacional e processos de aprendizagem; a Seção V aborda os elementos do jogo, como a história do personagem principal, mecânicas, conteúdos de aprendizagem e *gameplay*; na Seção VI são apresentadas as etapas do *design* experimental do jogo; a Seção VII discute os testes realizados e seus resultados e na Seção VIII, as considerações finais.

II. TRABALHOS RELACIONADOS

Pesquisas exploram a utilização de jogos como abordagens pedagógicas que auxiliem no desenvolvimento do Pensamento Computacional. Entre elas, o *Code Monkey* é um jogo educacional com proposta de aprendizagem baseada em tentativa e erro cujo objetivo é ajudar crianças, sem experiência prévia, a desenvolverem habilidades de programação [6].

Considerando que uma das formas de explorar o raciocínio lógico é através da matemática, o jogo *As Aventuras de Calculino* explora o raciocínio matemático através de desafios utilizando tabuada, labirinto e quebra-cabeça [7].

No mesmo sentido e abordando aspectos dos dois trabalhos anteriores, o jogo *Ninja Prog* tem como objetivo exercitar conceitos matemáticos vinculados à lógica de

programação. O jogador incorpora o personagem do Programador Ninja, que foi sequestrado, levado a um calabouço e só será libertado se resolver desafios ligados à manipulação de variáveis, conceitos de estrutura de repetição, operações aritméticas e desvios condicionais [8].

Assim como os trabalhos citados, *Looking for Pets* possibilita o desenvolvimento do Pensamento Computacional através da elaboração de algoritmos e apresenta como diferencial permitir a análise em diferentes dimensões do problema por meio da tecnologia de realidade aumentada e por forçar que o aluno pense no problema como um todo.

III. PENSAMENTO COMPUTACIONAL

Pensamento Computacional (PC) diz respeito a um conjunto de habilidades que permitem resolver problemas da melhor forma, sendo estas necessárias em todas as áreas do conhecimento e não somente nas áreas relacionadas à Computação [1]. Pesquisas mostram a crescente repercussão do PC no cenário da educação e seu potencial para aprendizagem [9], [4], [3].

Jogos que exercitam o Pensamento Computacional estão associados, normalmente, ao desenvolvimento de problemas em soluções fragmentadas, com relação direta à lógica de programação [10]. Esta característica vai ao encontro dos quatro pilares do PC: Decomposição, Reconhecimento de Padrão, Abstração e Algoritmos, sendo cada pilar uma etapa para o processo de resolução de um problema. A Decomposição é a divisão de um problema em partes menores. O Reconhecimento de Padrão é a identificação de similaridades entre características do problema. A Abstração refere-se a ignorar aspectos irrelevantes e focar somente no que é importante para a solução do problema e o Algoritmo é uma sequência finita de passos para gerar uma solução [11].

Diante do mencionado, o objetivo do jogo *Looking for Pets* é o desenvolvimento do Pensamento Computacional, tendo em vista seu potencial para análise e resolução de problemas através do raciocínio lógico, requisito fundamental para o século XXI [1].

IV. PROCESSO DE APRENDIZAGEM

Objetos de aprendizagem digitais têm atraído a atenção de pesquisadores nos últimos anos e exploram, principalmente, aspectos implícitos da aprendizagem [4].

O jogo *Looking for Pets* usa como base teórica da aprendizagem o Construcionismo de Seymour Papert. O mesmo defende que as pessoas podem aprender melhor a partir de suas experiências, sendo protagonistas em sua jornada de auto conhecimento ao criarem artefatos que explicitam ideias, sendo abstratas ou não. Esse processo preza pela auto regulação da aprendizagem em que o indivíduo pode aprender sozinho de acordo com os seus construtos [12], é possível encontrar semelhanças entre essa abordagem e o que a aprendizagem através de jogos pode proporcionar,

considerando que os jogos permitem ao indivíduos que busquem por soluções aos problemas propostos através das fases [13].

Na perspectiva construcionista as estruturas cognitivas do indivíduo são construídas e desenvolvidas sem a necessidade de um instrutor, podendo ser potencializadas através de uma ferramenta, como por exemplo, a *Linguagem Logo* (linguagem de programação visual para crianças desenvolvida em 1968). Assim como na linguagem LOGO, em *Looking for Pets* as crianças podem criar sequências de instruções para que o personagem execute uma ação capaz de solucionar o problema apresentado, ou seja, a criança primeiro cria o seu algoritmo e em seguida testa sua execução [14].

V. O JOGO *Looking for Pets*

Esta seção apresenta o jogo *Looking for Pets* que foi projetado para crianças com faixa etária de 10 a 13 anos, com o intuito de auxiliar no processo de aprendizagem de algoritmos e estimular as habilidades do Pensamento Computacional de forma divertida. *Looking for Pets* é um jogo de estratégia, desenvolvido na plataforma *Unity*, com códigos fontes baseados na linguagem de programação *C#*.

A. Processo de desenvolvimento

Nesta seção são apresentadas as etapas de desenvolvimento do jogo, bem como detalhes e informações a respeito de sua produção. As etapas do processo de desenvolvimento do jogo podem ser identificadas na Figura 1 e são descritas a seguir:

- **Idealização:** nesta etapa foram concebidas as primeiras ideias do jogo, incluindo a identificação dos principais elementos e características, tal como o formato de *gameplay* e mecânica, além de fazer um rascunho inicial da proposta visual. Como resultado desta etapa definiu-se o personagem principal e suas ações, a forma de pontuação através de estrelas, utilização de animais e como eles são distribuídos nas fases. Além disso, definiu-se que cada fase seria composta por caixotes. Ao discutir sobre o processo de transição entre fases, decidiu-se por colocar como objetivo o resgate de animais;
- **Elicitação e análise de requisitos:** foram levantados os requisitos para o desenvolvimento do jogo, entre eles os requisitos funcionais e não-funcionais, além dos aspectos pedagógicos e teoria cognitiva da aprendizagem. Durante o levantamento de requisitos foi discutida a garantia da auto regulação da aprendizagem no jogo, conduzindo o jogador a iniciar uma nova fase somente se a anterior estiver concluída, sempre aumentando a dificuldade da próxima fase. O resultado desta etapa é a lista dos requisitos funcionais e não-funcionais do jogo, desde o usuário ser capaz de executar os movimentos, até informações de disponibilidade e restrições. Os requisitos também passaram por uma

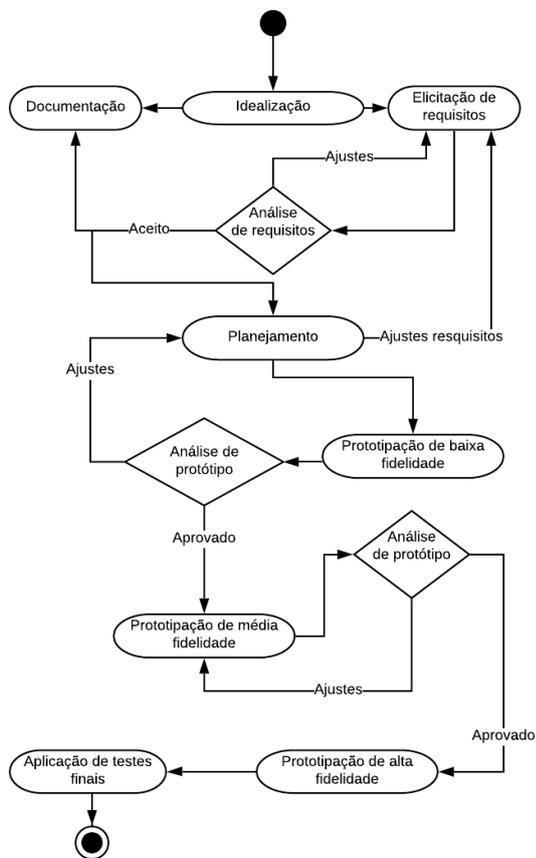


Figura 1. Etapas de desenvolvimento.

análise gerando as funcionalidades e restrições. Foram necessárias novas discussões que resultaram em novos requisitos e adaptações dos requisitos anteriores. Nas novas mudanças buscou-se ressaltar a importância e influência de cada requisito para o projeto final do jogo. Ao analisar as funcionalidades e requisitos, foi adicionada uma nova forma de pontuação, na qual o jogador além de resgatar os animais deve capturar estrelas (pontos) distribuídas pelas fases;

- **Planejamento:** nesta fase foi realizada uma adaptação do *Game Design Document* (GDD). Nele são apresentadas informações detalhadas do jogo, tais como: descrição de elementos da *gameplay* e mecânicas, características das fases, controles do jogador, sons e efeitos sonoros no ambiente, detalhes do *heads-up display* (HUD), entre outras. Por ser uma etapa que gera muitos artefatos, optou-se por descrevê-la melhor na seção V-B;
- **Protótipo de Baixa Fidelidade:** nesta etapa foi desenvolvido um modelo prévio de prototipagem visual do jogo, evidenciando a organização dos elementos nas

telas e funcionalidades. O resultado desta etapa é um pequeno protótipo que apresenta as principais telas do jogo, bem como o formato de *gameplay* e outros elementos importantes para mecânica, como por exemplo, controles, pontuação, objetivos e possíveis obstáculos. A (Figura 2) ilustra o primeiro protótipo desenvolvido para o jogo. Foi desenvolvido nas ferramentas *Microsoft Power Point* e *Adobe PhotoShop*. Após cada etapa de prototipação é feita a validação do protótipo, etapa descrita posteriormente ainda nesta seção;

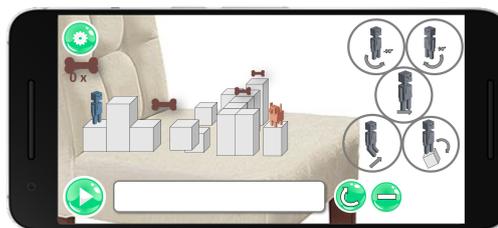


Figura 2. Primeiro protótipo do jogo.

- **Protótipo de Média Fidelidade:** este protótipo seguiu as diretrizes presentes no GDD durante a implementação do jogo, executado conforme informações descritas na documentação. Nesta etapa foi utilizada a *engine Unity* para a implementação. O resultado foi um protótipo mais maduro, que deu base para o projeto final do jogo. O protótipo constou com a implementação das funcionalidades do jogo, bem como seus requisitos discutidos e definidos em etapas anteriores. Nesta etapa o jogo ainda não possuía mecânicas em realidade aumentada. A (Figura 3) ilustra o protótipo de média fidelidade;



Figura 3. Segundo protótipo do jogo.

- **Protótipo de Alta Fidelidade:** consiste na implementação de melhorias no jogo, levando em consideração os resultados dos testes desenvolvidos nas etapas de Validação, em que é possível que as funcionalidades sejam ampliadas ou melhoradas. Neste momento o jogo passou a possuir mecânicas em realidade aumentada, além de novas mudanças em seu *design* visual e disponibilidade para dispositivos com sistema operacional Android. O resultado desta etapa pode ser visualizado na Figura 4;



Figura 4. Terceiro protótipo do jogo, com realidade aumentada.

- **Validação de protótipo:** a cada fim das etapas de protótipo de baixa, média e alta fidelidade o aplicativo é avaliado para verificar se as funcionalidades implementadas estão de acordo com os requisitos definidos. A validação após protótipo de baixa e média fidelidade é realizada pelos desenvolvedores. Após o protótipo de alta fidelidade é realizada a avaliação das interfaces utilizando o Teste de Usabilidade de Nielsen [15] que também foi executado pelos desenvolvedores. Os resultados de cada validação de protótipo são utilizados para melhorar ou corrigir aspectos no jogo. Os resultados dos testes estão descritos na Seção VII.
- **Aplicação de testes finais:** nesta etapa são realizados testes com o usuário. O aplicativo é validado com os usuários finais utilizando o teste MEEGA+KIDS [16], cujo objetivo é avaliar jogos com propósito educacional para aprendizagem de conceitos relacionados à Computação. Os resultados apresentam as diferentes perspectivas que os usuários têm em relação ao jogo, a ideia é analisar as métricas de avaliação de cada usuário. Os resultados das avaliações estão descritas na Seção VII.
- **Documentação:** durante as etapas do processo de desenvolvimento são gerados artefatos de documentação, entre eles o GDD. A documentação do projeto é composta de cronogramas de atividades dos desenvolvedores, bem como os requisitos e funcionalidades definidos em etapas anteriores.

B. Game Design Document - GDD

O GDD é um documento utilizado para registrar as principais etapas do desenvolvimento do jogo. Nele são descritas a história, a mecânica, as plataformas, entre outros. Apesar do GDD ser uma etapa no processo seguido, por sua relevância e artefatos gerados optou-se por descrever seus elementos nesta Seção, como segue:

1) *História:* o enredo do jogo narra a história de Lara, uma adolescente e estagiária de um centro comunitário de animais exóticos. O centro comunitário é afetado por tremores devido a erupção de um vulcão. Os animais ficam assustados com o abalo e alguns acabam fugindo, sendo o objetivo de Lara resgatar os animais desaparecidos.

2) *Fluxo do Jogo:* o *Looking For Pets* é um jogo do gênero *Puzzle*, envolvendo noções de elaboração de algo-

ritos, caracterizado pelo seu aspecto educacional, voltado para o desenvolvimento do raciocínio lógico. No jogo, os jogadores interpretam a pequena Lara cujo principal objetivo é resgatar todos os animais que acabaram fugindo por conta de terremotos gerados por um vulcão perto da cidade. O jogo possui 5 fases, cada fase possui um único animal que o jogador precisa alcançar. A dificuldade nas transições entre fases é perceptível pois a cada fase o grau de dificuldade aumenta gradativamente. O desempenho do jogador é medido através da quantidade de pontos que Lara alcança nas fases. A pontuação é representada por estrelas, sendo no máximo três, por fase.

3) *Mecânica:* para salvar os animais, o jogador deve propor uma sequência de ações (algoritmo) para fazer com que Lara chegue ao animal. Entre as ações possíveis estão ir para frente, virar 90° para a direita, virar 90° para a esquerda e pular. Após definidos os passos, o jogador seleciona a opção para executar o algoritmo e todos os passos são executados em uma única vez, seguindo a sequência definida pelo jogador. Para facilitar a visão do jogador é possível rotacionar a tela em um ângulo de 360° utilizando o *touchscreen* do dispositivo, espera-se, com isso, que gere mais imersão e diversão no jogador. A cada fase o jogador recebe estrelas que permitem desbloquear novos estilos visuais para Lara. Para uma partida iniciar, o dispositivo com o jogo instalado precisará localizar uma imagem de marcação, conforme ilustrada na Figura 5, e com isso é possível renderizar as fases e elementos da partida (Figura 6).



Figura 5. Imagem de marcação.

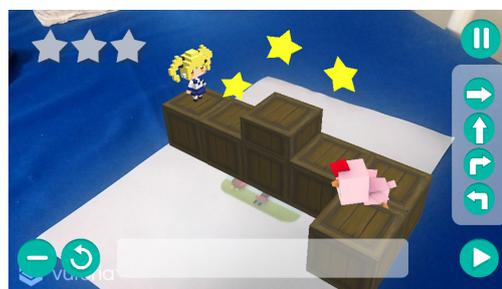


Figura 6. Fase renderizada.

Os jogadores precisam elaborar um algoritmo para cap-

turar cada animal desaparecido, distribuídos pelas fases do jogo. Caso o jogador não consiga alcançar o objetivo da fase, o personagem principal retorna para a sua posição inicial. A Figura 7 ilustra alguns dos animais utilizados no jogo.



Figura 7. Animais desaparecidos.

4) *Controles*: os jogadores utilizam botões para movimentação e outros comportamentos do personagem principal e estes estão localizados no lado direito da tela. Os comandos estão representados na Figura 8. Os movimentos possíveis para o personagem são: ir para a frente, Figura 8a, Pular, Figura 8b, virar para a esquerda, Figura 8c e virar para a direita, Figura 8g.

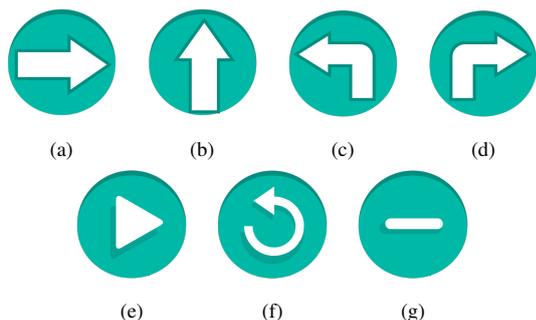


Figura 8. (a) Ir para a frente. (b) Pular. (c) Ir para a esquerda. (d) Ir para a direita. (e) Executar comandos. (f) Apagar todos os comandos. (g) Apagar último comando.

5) *Liberção de conteúdo*: Em cada fase do jogo o jogador pode conquistar até três estrelas sendo o desempenho dos jogadores determinado pelo número de estrelas capturadas. O jogo disponibiliza alguns estilos visuais para Lara, que podem ser trocados por estrelas conquistadas no final de cada fase. Os estilos disponíveis podem ser visualizados na Figura 9.



Figura 9. Estilos visuais de Lara.

6) *Telas*: A transição entre as telas pode ser visualizada no diagrama de telas, conforme ilustra Figura 10. É possível observar que, a partir do Menu Principal, o usuário pode ir para Configurações, Seleção de Fases ou Informações. É importante destacar que, levando em consideração a carga cognitiva da aprendizagem, o usuário só pode selecionar fases já visitadas. Caso ele ainda não tenha passado de uma fase, esta não estará habilitada para visita do usuário.

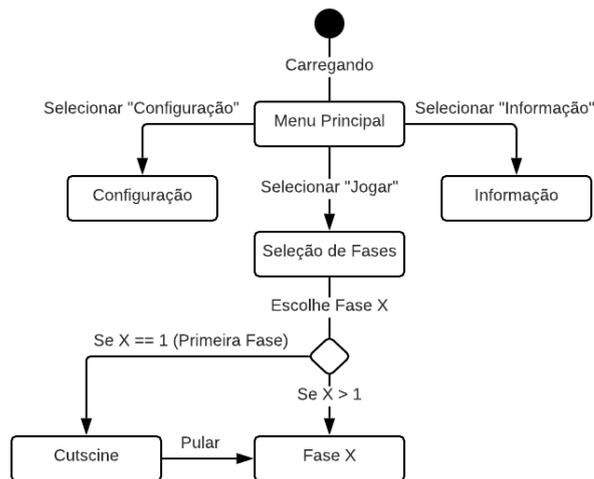


Figura 10. Diagrama de telas.

7) *Fases*: Cada fase do jogo possui um *layout* individual, se diferenciando através do percurso do jogador para alcançar seu objetivo. A Figura 11 ilustra a mecânica das cinco fases. É possível observar que a Fase 1, cujo objetivo é apresentar o cenário do jogo e ambientar o jogador, tem um nível de dificuldade e possibilidade de movimentos menores, conforme ilustra a Figura 11a. As Figuras 11b, 11c, 11d e 11e representam o caminho do jogador nas demais fases.

C. *Algoritmo no jogo: passo a passo*

Para desviar de obstáculos, conquistar estrelas (pontuação) e resgatar os animais de cada fase é necessário construir a solução mais adequada para cada etapa do jogo. Como exemplo, a Figura 12 apresenta na imagem 12a, a tela da Fase 3 do jogo, ilustrando antes de começar a partida e uma possível solução para a Fase 3, na imagem 12b. Após o jogador selecionar o conjunto de movimentos que representa a solução, este executa e caso esteja correto vai para a próxima fase, caso contrário volta para o início da fase atual.

D. *Pensamento Computacional e o jogo*

O jogo busca exercitar as habilidades do Pensamento Computacional induzindo os jogadores a criarem soluções criativas através da construção de algoritmos. A Tabela I apresenta a relação dos pilares do PC com aspectos do jogo.

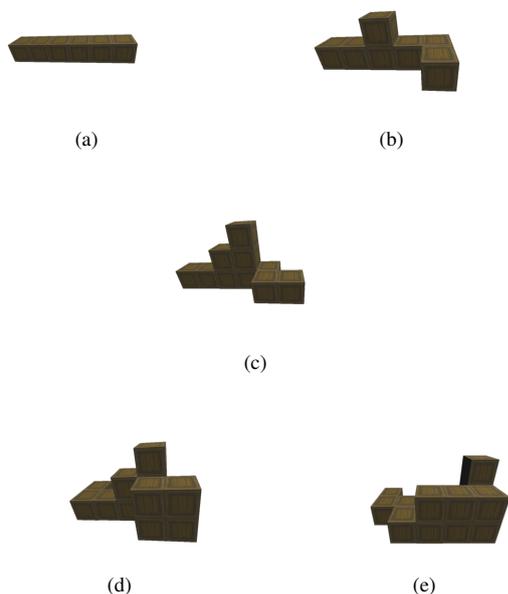


Figura 11. (a) Layout da Fase 1. (b) Layout da Fase 2. (c) Layout da Fase 3. (d) Layout da Fase 4. (e) Layout da Fase 5.

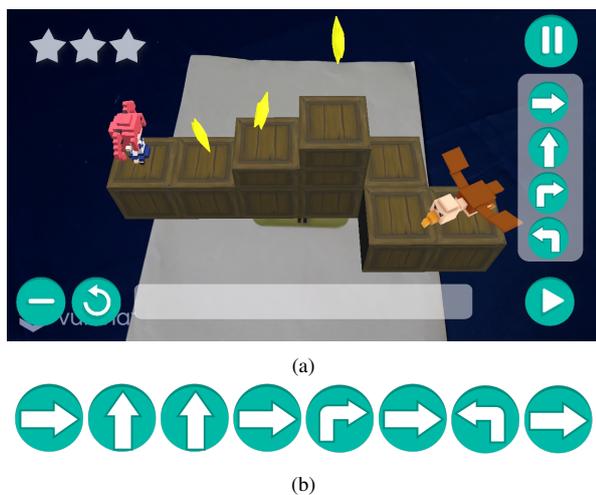


Figura 12. (a) Fase 3 do Jogo. (b) Possível solução para a Fase 3.

VI. CENÁRIO DE AVALIAÇÃO

Esta seção apresenta os tipos testes e cenário de avaliação. Foram realizados duas classes de teste, com desenvolvedores e com o público-alvo. Na primeira classe, foram avaliados os elementos de objetos de aprendizagem [17] e a usabilidade, através do teste de heurística de Nielsen, que analisa a qualidade da interface do jogo por meio de perguntas heurísticas [15]. Estes dois testes foram realizados pelos projetistas.

No segundo momento, foi avaliada a percepção dos usuários, utilizando o método MEEGA+KIDS. Ele é usado

Tabela I. Pensamento Computacional e *Looking for Pets*.

Pensamento Computacional	<i>Looking for Pets</i>
Decomposição	Organizar de forma sistemática os elementos e objetos em uma fase
Reconhecimento de padrões	Organização das fases; Localização de botões; Distribuição dos elementos do jogo;
Abstração	Principais objetivos do jogo: - Posição dos objetivos de cada fase; - Quantidade de pontos possível de capturar; - Formas de locomoção do personagem;
Algoritmo	Sequência de passos para: - Capturar objetivos; - Coletar pontos; - Contornar obstáculos;

para validar os aspectos de usabilidade e experiência do jogo, sendo uma ferramenta para avaliar jogos educacionais cujo público-alvo é infantil [16]. O método MEEGA+KIDS propõe 12 categorias, sendo: Estética, Aprendizagem, Controle, Acessibilidade, Desafio, Satisfação, Interação Social, Diversão, Atenção, Relevância, Confiança, Aprendizagem Percebida [16]. Como o jogo proposto não aborda aspectos de acessibilidade e interação social, estas categorias foram retiradas da avaliação, os tópicos Confiança e Aprendizagem Percebida também não foram avaliados nesse processo, pois as crianças demonstraram cansaço na fase de entrevistas.

A avaliação foi aplicada em dois grupos de pessoas: Grupo A, composto por 10 estudantes do ensino fundamental com faixa etária entre 10 e 13 anos e Grupo B, composto por 7 estudantes de ensino superior de cursos da área de Computação. A proposta de realizar o teste com os dois grupos é verificar se existe diferença da análise entre crianças, que desconhecem os conceitos formais de algoritmos e adultos ligados à área de Computação, que compreendem os conceitos formais e suas estrutura de dados. Caso as crianças se saíssem bem, poderia apontar para a aceitação de Wing, no que diz respeito ao Pensamento Computacional ser inato.

O teste MEEGA+KIDS avalia em escalas entre 0 a 4, sendo: 0 = Não possui, 1 = Existem alguns, 2 = Suficiente, 3 = Bom, 4 = Ótimo. A escala numérica foi representada por figuras popularmente conhecidas como "Emojis". A Tabela II apresenta a descrição dos critérios analisados.

VII. RESULTADOS E DISCUSSÕES

Nesta sessão são apresentados os resultados e as análises dos testes realizados com os projetistas do jogo e com os usuários.

A. Resultado da avaliação com desenvolvedores

Para avaliar os requisitos de jogos, o *Looking for pets* foi avaliado seguindo os critérios de existência ou não de

Tabela II. Tabela de critérios selecionados do MEEGA+KID.

Critérios	Descrição da avaliação
Estética	Se a interface do jogo está agradável ao usuário.
Aprendizagem	Se o jogo pode ser utilizado pelo usuário para alcançar metas específicas de aprender a jogar com eficiência, satisfação e liberdade.
Controle	Se o jogo pode ser usado por usuários com problemas visuais.
Desafio	Se o jogo possui grau de dificuldade adequado ao desempenho do usuário.
Satisfação	Se os alunos sentem que o esforço no jogo resulta em algum tipo de aprendizado.
Diversão	Se o jogo promove sensações de prazer, felicidade, entretenimento e distração nos usuários.
Atenção	A atenção, concentração e foco.
Relevância	A relação da percepção dos usuários com a proposta educacional.

elementos de jogos [17]. São 29 critérios divididos em 3 grupos: elementos de dinâmica, mecânica e componentes, sendo o teste aplicado ao grupo de projetistas do jogo formado por três pessoas. A Tabela III apresenta os critérios que o jogo cumpre.

Tabela III. Resultados da avaliação do enquadramento [17].

Critérios	Descrição
Elementos de dinâmica do objeto de aprendizagem	100%
Mecânica do objeto de aprendizagem	55.5%
Componentes do objeto de aprendizagem	57.1%

A heurística de Usabilidade de Nielsen foi aplicado com três avaliadores com experiência em desenvolvimento e análise de objetos de aprendizagem. O teste é composto por 10 heurísticas e busca avaliar o *design* de interface do usuário. A Tabela IV apresenta dos resultados da aplicação do teste.

A heurística "Controle do usuário e liberdade" apresenta a maior média entre os critérios avaliados com um valor de 96,67%, sendo a "Visibilidade do Estado do sistema" o segundo maior valor com 84,44%. O teste revela resultados positivos quanto ao *design* de interface do usuário do jogo.

B. Resultado da avaliação com os usuários

O método MEEGA+KIDS foi aplicado com a finalidade de avaliar o jogo durante fase de protótipo. A Tabela V apresenta o resultado do teste com o Grupo A e a avaliação de cada critério. Os resultados dos critérios analisados em um intervalo de Bom e Ótimo possuem índices bem avaliados. Destaque para o critério Estética com 95% e Satisfação com 83.4% de aceitação. Grande parte dos critérios avaliados pelo Grupo A recebeu notas altas, evidenciando a relevância dos elementos do jogo. Já no grupo B a avaliação teve o intuito de analisar uma possível associação com os resultados do Grupo A, gerando evidências para traçar uma

Tabela IV. Resultados do teste de Usabilidade de Nilsen.

Heurísticas	Nota 1	Nota 2	Nota 3	Média
Visibilidade do Estado do sistema	80%	87%	87%	84,44%
Correspondência entre o sistema e o mundo real	80%	50%	70%	66,67%
Controle do usuário e liberdade	90%	100%	100%	96,67%
Consistência e Uso de padrões	90%	70%	85%	81,67%
Prevenção de erro	0%	40%	20%	20%
Reconhecimento em vez de memorização	50%	50%	50%	50%
Ajude os usuários a reconhecerem, diagnosticarem e recuperarem-se de erros	86,67%	86,67%	93,33%	88,89%
Eficiência e flexibilidade de uso	48,57%	54,29%	77,14%	60%
Estética e design minimalista	40%	100%	80%	73,33%
Ajuda e documentação	80%	28%	50%	52,78%

Tabela V. Resultado da avaliação MEEGA+KIDS (Grupo A).

Critérios	Não possui	Existem alguns	Suficiente	Bom	Ótimo
Estética	0%	0%	15%	40%	45%
Aprendizagem	10%	5%	25%	20%	40%
Controle	0%	5%	15%	25%	55%
Desafio	0%	0%	16.6%	36.7%	46.5%
Satisfação	0%	0%	16.6%	33.4%	50%
Diversão	0%	5%	15%	25%	55%
Atenção	0%	5%	0%	30%	65%
Relevância	0%	6%	22.5%	46.5%	25%

análise comparativa entre os perfis. A Tabela VI apresenta os resultados obtidos em cada critério do Grupo B.

Os resultados da avaliação MEEGA+KIDS, no Grupo B, também apresentam a maioria das avaliações no intervalo de Bom e Ótimo, destacando critérios como: Estética com 92.8%, Controle com 92.8% e Desafio possuindo 100%. A Figura 13 apresenta a comparação dos resultados dos grupos avaliados.

A Figura 14 apresenta a distribuição geral em porcentagem das métricas analisadas pelos Grupos A e B. Na

Tabela VI. Resultado da avaliação MEEGA+KIDS (Grupo B).

Critérios	Não possui	Existem alguns	Suficiente	Bom	Ótimo
Estética	0%	0%	7.2%	42.8%	5%
Aprendizagem	0%	0%	38.5%	46.2%	15.4%
Controle	0%	0%	7.2%	42.8%	5%
Desafio	0%	0%	0%	33.4%	66.6%
Satisfação	0%	0%	14.3%	47.6%	38.1%
Diversão	0%	0%	3.4%	46.3%	50.3%
Atenção	0%	7.1%	28.6%	36.7%	27.6%
Relevância	0%	0%	3.4%	27.8%	68.8%

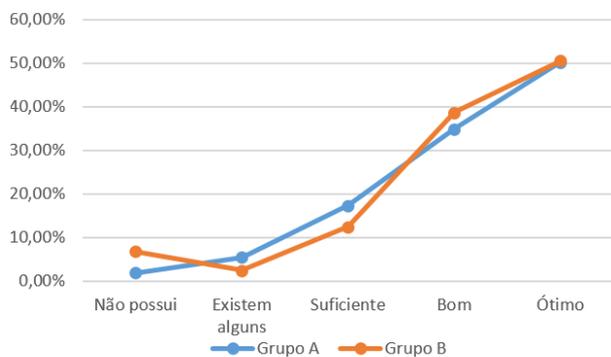


Figura 13. Média de porcentagem do Grupo A e Grupo B.

Figura 14 é possível observar a predominância de "Ótimo" e "Bom". Apesar do baixo número de participantes no teste, as notas dos grupos apresentam um padrão em relação às métricas analisadas no jogo, em "Estética", "Controle" e "Diversão", por exemplo, é possível observar notas equivalentes. Os grupos, divididos em estudantes de Computação (com afinidade em conceitos de algoritmos) e crianças da educação infantil (sem contato com conceito formal), conseguiram progredir no jogo, mesmo com participantes em contextos diferentes, os grupos realizaram observações equivalentes e conseguiram exercitar e estimular as habilidades do Pensamento Computacional presentes no jogo.

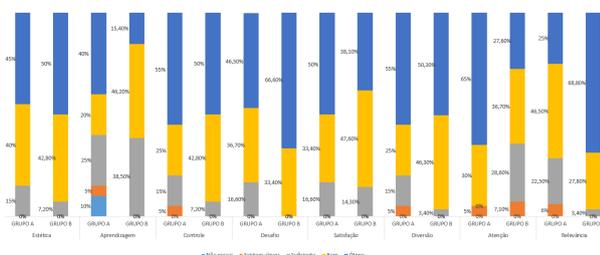


Figura 14. Visão geral de notas do Grupo A e B.

VIII. CONSIDERAÇÕES FINAIS

Este trabalho apresentou o desenvolvimento e aplicação, do jogo *Looking for Pets* como alternativa para exercitar as habilidades do Pensamento Computacional, através da construção de algoritmos de forma lúdica.

Para o desenvolvimento foi utilizada uma sequência de etapas bem desenhadas e definidas, entre elas, a construção do GDD que gerou artefatos importantes para a implementação do jogo.

O jogo, apesar de ter sido proposto para crianças com idade entre 10 e 13 anos, pode ser usado para público adulto. Resultados de avaliações realizadas os desenvolvedores revelam que o jogo é uma ferramenta com potencial

para aprendizagem de algoritmos e desenvolvimento do raciocínio lógico.

Como trabalhos futuros se aponta o desenvolvimento de novas fases bem como uma versão para utilização colaborativa, através de lousa digital interativa.

REFERENCES

- [1] J. M. Wing, "Computational thinking," *Communications of the ACM*, vol. 49, no. 3, pp. 33–35, 2006.
- [2] C. Brackmann, R. M. Boucinha, M. Roman-Gonzalez, D. A. C. Barone, and A. Casali, "Pensamento computacional desplugado: Ensino e avaliação na educação primária espanhola," in *Anais dos Workshops do Congresso Brasileiro de Informática na Educação*, vol. 6, no. 1, 2017, p. 982.
- [3] J. M. Wing and D. Stanzione, "Progress in computational thinking, and expanding the hpc community," *Communications of the ACM*, vol. 59, no. 7, pp. 10–11, 2016.
- [4] D. Melo, F. G. de Sousa Pires, R. Melo, and R. J. d. R. S. Júnior, "Robô euro: Game de estratégia matemática para exercitar o pensamento computacional," in *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, vol. 29, no. 1, 2018, p. 685.
- [5] T. Fullerton, *Game Design Workshop A Playcentric Approach to Creating Innovative Games*. Nova York: Taylor and Francis Group, LLC. CRC Press, 2019.
- [6] J. Godinho, G. Lima, N. Santos, J. Penna, J. Gomide, and G. Barbosa, "Apreciação da usabilidade em jogos educativos de programação voltados para crianças," in *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, vol. 29, no. 1, 2018, p. 1888.
- [7] E. J. S. Batista, J. Mioto, C. A. C. Bogarim, A. Lima, and Q. Araujo, "As aventuras de calculino: jogo para ensino de raciocínio lógico," in *Anais dos Workshops do Congresso Brasileiro de Informática na Educação*, vol. 6, no. 1, 2017, p. 451.
- [8] A. Raabe, A. M. Santana *et al.*, "Ninja prog-jogo de conceitos matemáticos e lógica de programação," in *Anais dos Workshops do Congresso Brasileiro de Informática na Educação*, vol. 5, no. 1, 2016, p. 159.
- [9] V. Barr and C. Stephenson, "Bringing computational thinking to k-12: what is involved and what is the role of the computer science education community?" *Inroads*, vol. 2, no. 1, pp. 48–54, 2011.
- [10] T. J. Medeiros, T. R. da Silva, and E. H. da Silva Aranha, "Ensino de programação utilizando jogos digitais: uma revisão sistemática da literatura," *RENOTE*, vol. 11, no. 3, 2013.
- [11] L. Liukas, *Hello Ruby: adventures in coding*. Macmillan, 2015, vol. 1.
- [12] S. Papert, *Mindstorms: Children, computers, and powerful ideas*. Nova York: Basic Books, Inc., 1980.

- [13] J. P. Gee, “What video games have to teach us about learning and literacy,” *Computers in Entertainment (CIE)*, vol. 1, no. 1, pp. 20–20, 2003.
- [14] S. Papert, “What is logo? who needs it,” *Logo philosophy and implementation*, 1999.
- [15] J. Nielsen, *Usability Engineering*, ser. Interactive Technologies. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, 1994. [Online]. Available: <https://books.google.com.br/books?id=DBOowF7LqIQC>
- [16] C. G. von Wangenheim, G. Petri, and A. F. Borgatto, “Meega+kids: A model for the evaluation of educational games for computing education in secondary school,” 2018.
- [17] F. Alves, *Gamification: Como criar experiências de aprendizagem engajadoras*. São Paulo: DVS Editora, 2015. [Online]. Available: <https://books.google.com.br/books?id=JnOwDQAAQBAJ>