

As Aventuras de Salazar: um *serious game* para desenvolvimento do Pensamento Computacional

Elisa Boff* e Henrique Pagno de Lima†

Resumo

Este artigo apresenta a modelagem, o desenvolvimento e a avaliação de um *serious game* para auxiliar crianças a desenvolverem o pensamento computacional. Os *serious games* tem se tornado cada vez mais populares nos dias de hoje. Isso ocorre principalmente porque proporcionam uma forma divertida e engajante de aprendizagem, trabalhando como um auxílio de ensino para aqueles que o usufruem. O *serious game* proposto destina-se a um público-alvo específico de seis a oito anos de idade, visando ensinar aos mesmos os conceitos básicos desta forma de pensamento. A seleção desta faixa etária baseou-se na Teoria Cognitiva de Piaget, que estuda a evolução cognitiva do ser humano ao longo do seu desenvolvimento. A metodologia para desenvolvimento seguiu o framework DPE (Design, Play, Experience) estruturar o protótipo do *serious game* proposto, dividindo-o nas camadas apresentadas pelo framework. O framework é organizado de forma que o objetivo de ensino do *serious game* seja o foco na criação do mesmo, não permitindo que este seja tomado como segundo plano no projeto. Concluindo, o protótipo de *serious game* proposto neste trabalho é direcionado a plataformas *mobile* e objetiva ensinar a crianças os conceitos básicos de lógica para computação de uma forma divertida e interessante.

Palavras-chave

Serious games. Jogos *mobile*. Pensamento computacional. Framework DPE. Lógica computacional.

The Adventures of Salazar: a serious game for Computational Thinking development

Abstract

This article presents the modeling, development and evaluation of a serious game to help children develop computational thinking. Serious games have become increasingly popular these days. This is primarily because they provide a fun and engaging way of learning, working as a teaching aid for those who enjoy it. The proposed serious game is intended for a specific target audience of six to eight years old, aiming to teach them the basic concepts of this way of thinking. The selection of this age group was based on Piaget's Cognitive Theory, which studies the cognitive evolution of human beings throughout their development. The methodology for development followed the DPE (Design, Play, Experience) framework to structure the prototype of the proposed serious game, dividing it into the layers presented by the framework. The framework is organized in such a way that the serious game teaching objective is the focus on its creation, not allowing it to be taken as a background in the project. In conclusion, the serious game prototype proposed in this work is aimed at mobile platforms and aims to teach children the basic concepts of logic for computing in a fun and interesting way.

Keywords

Serious games. Mobile games. Computational thinking. DPE Framework. Computational logic.

I. INTRODUÇÃO

Serious games podem ser definidos como jogos que não têm entretenimento, prazer, ou diversão como seu principal propósito [1].

Os *serious games* têm se tornado, ao longo dos últimos anos, cada vez mais atraentes. Em parte, porque, com o avanço da tecnologia, os dispositivos eletrônicos se tornaram muito acessíveis à população como um todo. Além disso, pelos estímulos que os jogos geram aos usuários. A diversão que os games proporcionam advém do fato de aprender a superar um desafio, de dominar uma nova tarefa, de compreender um novo quebra-cabeça [2]. Jovens de todas as idades estão

acostumados com jogos, sejam eles eletrônicos ou não, em seu dia a dia e durante seu crescimento. São raros aqueles que não tem um smartphone em mãos ou simples acesso a um computador. Isso proporciona uma oportunidade para o uso dos mesmos no auxílio do seu desenvolvimento acadêmico e pessoal, afinal, as novas gerações da sociedade possuem uma grande afinidade com a tecnologia. O que não se pode negar é a forte influência que esse contexto tecnológico exerce nas atividades humanas [2].

Tornar a aprendizagem agradável e prazerosa é, sem dúvida, a melhor forma de estimulá-la. No entanto, é preciso avaliar a melhor forma na qual um *serious game* deve ser desenvolvido

*Programa de Pós-Graduação em Ensino de Ciências e Matemática, Universidade de Caxias do Sul; †Curso de Ciência da Computação, Universidade de Caxias do Sul.

E-mail: eboff@ucs.br

Data de envio: 19/12/2022

Data de aceite: 22/12/2022

<http://dx.doi.org/10.18226/23185279.v10iss1p23>

para o estímulo do conhecimento, não interferindo no seu real objetivo. Ou seja, como se pode criar um ambiente de aprendizagem estimulante e eficiente.

O estudo de lógica para computação orientado para crianças é de muita utilidade para elas, uma vez que será aproveitado independente da área e profissão escolhida para cursar e exercer ao longo da vida adulta. Assim, o objetivo principal do trabalho é a modelagem, desenvolvimento e avaliação de um *serious game* que auxilie crianças a desenvolverem lógica computacional.

II. PENSAMENTO COMPUTACIONAL E *SERIOUS GAMES*

Para Wing [3], professora do departamento de Ciência da Computação na universidade de Carnegie Mellon, o Pensamento Computacional é o uso do raciocínio heurístico para descobrir uma solução. É planejar, aprender, e dimensionar na presença da incerteza.

O pensamento computacional está diretamente relacionado ao desenvolvimento da capacidade de pensarmos em soluções para variados problemas, sejam eles acadêmicos ou não. A lógica que desenvolvemos pode ser utilizada para otimizar o tempo nas prateleiras do supermercado, para escolher qual caixa será o mais rápido, ou para compreendermos com mais facilidade conceitos acadêmicos não-relacionados à computação. Jonassen [4] complementa este raciocínio, afirmando que aprender a resolver problemas é a habilidade mais importante que um estudante pode desenvolver, em qualquer situação.

Pensamento computacional é usar abstração e decomposição ao enfrentar uma tarefa grande e complexa ou projetar um sistema grande e complexo. É a separação de preocupações. É escolher uma representação apropriada para um problema ou modelar os aspectos relevantes de um problema para torná-lo tratável. É usar invariantes para descrever o comportamento de um sistema de forma sucinta e declarativa. É ter a confiança de que podemos, de forma segura, usar, modificar e influenciar um sistema grande e complexo sem compreender todos os seus detalhes [3].

Wing [3] vai mais longe ainda, ao utilizar conceitos específicos do pensamento computacional para descrever sua utilidade em tarefas cuja relação com a computação não é diretamente aplicável. A abstração e decomposição de tarefas para a sua simplificação são métodos comuns na computação, que podem ser aplicados em qualquer atividade que tenha maior complexidade.

O pensamento computacional é uma habilidade fundamental para todos, não apenas para cientistas computacionais. Além de leitura, escrita e aritmética, deveríamos adicionar o pensamento computacional para as habilidades analíticas de todas as crianças.

Introduzir o conhecimento do pensamento computacional no início dos estudos de jovens é introduzi-los aos benefícios desta forma de pensamento. A solução bem-sucedida de problemas requer que estudantes gerem e testem soluções em sua mente antes de testá-las no mundo físico [4]. O desenvolvimento da lógica pode auxiliar em todas as tarefas e desafios que sejam enfrentados ao longo do crescimento acadêmico e pessoal, uma vez que estes podem ser visualizados de forma mais linear, separando-os em pequenos

passos para encontrar uma solução. Quando estudantes estão solucionando problemas, eles aprendem e compreendem mais [4].

Os métodos de ensino deste pensamento, no entanto, não são consolidados. Ao menos, não para crianças. É importante que os estudantes possam compreender, de fato, os conceitos do pensamento computacional, e não apenas como aplicá-los em sua ferramenta predominante, o computador. Um bom exemplo disso é o uso da calculadora, que pode afetar a compreensão de cálculos caso seja introduzida durante os primórdios do ensino de aritmética. É discutível também, qual seria uma ordenação adequada para a aprendizagem dos conceitos por crianças. Aprendemos números aos cinco anos de idade, álgebra aos 12 anos, e cálculo a partir dos 18. Podem haver muitas formas de estruturar a progressão dos conceitos de pensamento computacional; qual é o método mais eficiente para cada tipo de estudante? [5]

O estudo para definição da faixa etária para qual o jogo digital foi desenvolvido passou pela compreensão da teoria da Epistemologia Genética de Jean Piaget [6] para compreender como os jovens, de diferentes idades, respondem a estímulos e novas experiências ao longo de seu desenvolvimento cognitivo. A teoria de Piaget esclarece como o crescimento intelectual humano ocorre ao longo dos primeiros anos de vida. Piaget discordava da ideia de que a inteligência é uma característica fixa, e considerava o desenvolvimento cognitivo como um processo que ocorre devido à maturação biológica e a interação com o ambiente [6]. Assim, Piaget mostrou que jovens pensam de formas diferentes quando comparados com adultos. Antes de sua pesquisa, acreditava-se que crianças eram apenas pensadores menos competentes que adultos. De acordo com Piaget, crianças nascem com uma estrutura mental muito básica (geneticamente herdada e evoluída) na qual se baseia todo o aprendizado e conhecimento subsequente [6]. Cada criança passa pelos estágios na mesma ordem e o seu desenvolvimento é determinado por maturação biológica e a sua interação com o meio ambiente. O estágio Operacional Concreto, que se encontra na faixa de 7 a 11 anos, é considerado por Piaget como um marco no desenvolvimento cognitivo infantil, visto que ele marca o início do pensamento lógico ou operacional. Assim, neste estágio, a criança é capaz de raciocinar e aplicar soluções em sua cabeça (realizar abstrações), ao invés de precisar observá-las fisicamente no mundo real. Desta forma, para fins de delimitação deste projeto, definiu-se como faixa etária ideal o final do estágio Pré-operacional (2 a 7 anos) e o início do estágio Operacional Concreto (7 a 11 anos), idade em que ocorre o início do pensamento lógico.

Além da Teoria Cognitiva de Piaget, é oportuno abordar na fundamentação deste trabalho, a Zona do Desenvolvimento Proximal (ZDP) de Vygotsky. A ZDP é a distância entre o nível de desenvolvimento real, que se costuma determinar por meio da solução independente de problemas, e o nível de desenvolvimento potencial, determinado por meio da solução de problemas sob a orientação de um adulto ou em colaboração com companheiros mais capazes [7]. O desenvolvimento real é aquele cujos processos de desenvolvimento encontram-se já consolidados. São atividades que são realizadas por um indivíduo sem auxílio de

um terceiro, uma vez que a pessoa já é capaz de solucionar a necessidade por conta própria. O desenvolvimento potencial caracteriza aquilo que um indivíduo é capaz de realizar somente com algum tipo de ajuda, advinda de outro mais capaz ou mais experiente. São atividades que um indivíduo ainda não consegue solucionar por conta própria, uma vez que ainda não tem o total conhecimento necessário para executá-la. Vygostky defendia que, por meio desta abordagem, era possível melhor estimar o atual nível de desenvolvimento de uma criança, pois ele levava em consideração aquelas funções que, apesar de não estarem completamente desenvolvidas, encontravam-se presentes no indivíduo.

Convém ressaltar, portanto, que um *serious game* pode ser utilizado como o “auxílio” mencionado durante o nível de desenvolvimento potencial, uma vez que este pode agir como o facilitador para a aprendizagem de um conceito, tema ou teoria. Ou seja, nesse caso, os games podem proporcionar a interação com o outro da perspectiva de Vygotsky. Isso significa que, ao interagir com um *serious game*, a criança recebe auxílio até o momento em que ele consegue resolver os problemas sem assistência.

Os *serious games* são jogos que têm uma finalidade educativa explícita e cuidadosamente pensada, não se destinando a serem jogados principalmente pelo entretenimento. Isso não significa que os *serious games* não são, ou não deveriam ser, divertidos [8].

Laamarti et al. [1] também observam que muitos jogos eletrônicos, que não são caracterizados como *serious games*, podem apresentar oportunidades de aumento de conhecimento ou habilidade em relação à alguma atividade ou assunto. Podemos exemplificar estes casos com jogos de estratégia, como a série Total War, ou o simulador de corridas Gran Turismo, que podem agregar conhecimentos de história e estratégia, ou desenvolver reflexos e habilidades que podem ser utilizados em atividades diárias. Jogos em geral também podem aprimorar o conhecimento e habilidades linguísticas dos usuários. Estudantes do idioma inglês, por exemplo, que usufruem de jogos, podem utilizá-los como um treinamento adicional de suas habilidades. Assim, é importante ressaltar que a diferença de um *serious game* para um jogo comercial da indústria está na ênfase que lhe é adicionado para o objetivo principal de desenvolvimento de algum conhecimento, e não para o entretenimento em si. Este conhecimento está relacionado ao conceito específico do *serious game*, como bem-estar, educação e saúde.

Tabela 1 - Habilidades do Pensamento Computacional

Habilidade	Descrição	Exemplo
Lógica condicional	Consiste na habilidade de compreender as consequências dos valores verdadeiro e falso. Usualmente através da utilização da construção <i>se-então-senão</i> .	Dados 3 números x, y e z. Se x é maior que y e y é maior que z. Então, sabemos que x é o maior. Senão, o maior valor será y ou z. (tabela verdade).
Construção de algoritmos	Consiste na habilidade de resolver um determinado	Dados 3 números x, y e z. Primeiro verifica-se se x é maior que y e

	problema, utilizando um conjunto de lógicas condicionais em uma abordagem passo a passo.	z. Se for verdade, então x é o maior. Senão, preciso verificar se y é maior que z. Se for verdade então y é maior. Se não z é o maior.
Depuração	Consiste no ato de encontrar erros lógicos em um algoritmo que não funciona como esperado.	O algoritmo pode se não comportar como esperado para uma determinada entrada. Em sala de aula é explorada através de atividades que envolvam encontrar um erro dentro de um algoritmo.
Simulação	Consiste no ato de modelar ou testar um algoritmo. São usadas tanto para a depuração quanto para a construção de algoritmos.	Dado um algoritmo que retorna o maior valor, podemos simular o algoritmo para diferentes combinações de valores de x, y e z. Em sala de aula é explorada através de teste de mesa e execuções passo a passo.
Socialização	Refere aos aspectos sociais do pensamento computacional. Onde a solução de um problema pode ser alcançada e compartilhada entre uma ou mais pessoas.	A solução de um problema pode ser alcançada pela divisão e distribuição do problema para uma ou mais pessoas. Dojo de programação é um exemplo de metodologia que enfatiza essa habilidade (SATO et al., 2008 apud COSTA, SOUZA, et al., 2015). Ela enfatiza a programação em pares que codificam e compartilham com todos as suas soluções.

Tabela 2 – Análise dos jogos com base nas habilidades do Pensamento Computacional desenvolvidas

Nome	Gênero	Plataforma	Habilidades	Forma de Iteração	Faixa etária
CodeCombat	Luta	Web	Lógica condicional, Construção de algoritmos, Depuração, Socialização	Texto	10+
CodinGame	Aventura	Web	Lógica condicional, Construção de algoritmos	Texto	10+

			os, Depuraç ão, Socializ ação		
Robo Logic	Quebra- cabeças	iOS	Constru ção de Algorit mos, Simulaç ão	Visual	5 a 8 anos
Lightb ot	Quebra- cabeças	Web, iOS, Android	Constru ção de Algorit mos, Simulaç ão	Visual	

Existem vários jogos no mercado para desenvolver a lógica de programação. Entre eles, podemos citar: CodeCombat, CodinGame, RoboLogic e LightBot.

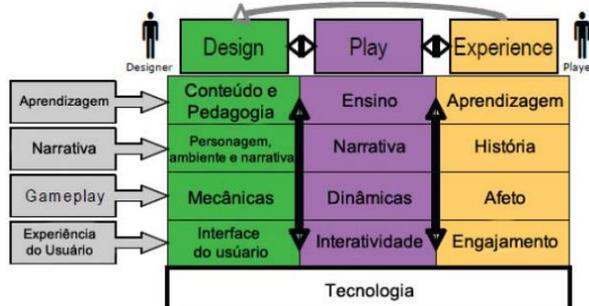
Codecombat é um *serious game* de RPG multiplayer, cujo propósito é a aprendizagem de programação para crianças. O CodinGame, semelhante ao CodeCombat, permite programar soluções para problemas introduzidos dentro de um jogo. O RoboLogic é um jogo mobile onde o jogador deve mover um robô através de um cenário, ativando caixas específicas do trajeto para completar cada nível. O LightBot tem um propósito semelhante ao RoboLogic, mas difere-se por estar disponível para várias plataformas móveis.

Estes jogos digitais podem ser analisados com base nas habilidades do Pensamento Computacional que propõe desenvolver. A Tabela 1 apresenta as habilidades do Pensamento Computacional, segundo Costa et al [9]. A Tabela 2, por sua vez, apresenta nossa análise dos serious games citados nesta seção com base na Tabela 1.

III. O *SERIOUS GAME* AS AVENTURAS DE SALAZAR

Conforme Winn [5], para projetar um jogo de forma eficiente, o designer deve antes definir os objetivos para a experiência que se deseja alcançar. Estes objetivos podem ser usados tanto para guiar o projeto quanto para medir a eficiência do mesmo quando implementado.

Figura 1 – Estrutura do framework DPE [10]



Ainda conforme Winn [5], é necessário termos ciência de que jogar é uma experiência mediada. Esta experiência é muito influenciada pelo jogador, sua capacidade de cognição, sociabilidade, cultura e suas experiências de vida. Assim, a experiência de um jogador pode ser profundamente diferente da experiência de outros jogadores. Portanto, o público-alvo para o jogo deve ser considerado durante o processo de projeto do mesmo.

A Figura 1 apresenta o framework DPE, utilizado na modelagem do jogo proposto. Este framework descreve os subcomponentes de design de um *serious game* e sua conexão com os componentes principais, incluindo a aprendizagem, narrativa, gameplay e experiência do usuário. A tecnologia é mencionada abaixo da estrutura, uma vez que o projeto do jogo é realizado na mesma.

A. Camada de Aprendizagem

A camada de aprendizagem diz respeito ao conteúdo e à pedagogia, visando resultar na aprendizagem do jogador enquanto o mesmo joga o jogo. O conteúdo e a pedagogia do jogo devem ser projetados com o intuito de alcançar um objetivo de aprendizagem resultante da experiência do jogo. É importante definir, no início do projeto, os objetivos de aprendizagem que serão introduzidos no jogo. Estes objetivos também formam a base para a avaliação da eficiência de ensino que o jogo provém aos jogadores. Desta forma, o designer do jogo deve definir os objetivos da aprendizagem para que, posteriormente, possa elaborar os conteúdos para atender a esses objetivos.

Winn [5] sugere a Taxonomia de Bloom para Ensinar e Aprender [11] como recurso para estruturar e gerar os resultados de aprendizagem de um *serious game*. Este trabalho, no entanto, é apoiado nas teorias de Zona de Desenvolvimento Proximal de Vygostky e na Teoria Cognitiva de Piaget (Seção 2) para a estrutura da aprendizagem do tema em questão.

A camada de aprendizagem deste protótipo, assim como o Robo Logic e o Lightbot, visa apresentar aos jogadores os conceitos básicos da construção de um algoritmo e a simulação do mesmo. Uma vez que o público proposto está em seus primeiros contatos com o pensamento computacional, a aprendizagem que devemos propor aos mesmos é simples e de fácil compreensão.

Assim, definiu-se que este *serious game* deveria apresentar ao jogador algoritmos sequenciais, o conceito de *loops* e de procedimentos. O jogador monta uma sequência de comandos (ou seja, um algoritmo) para mover o personagem e atingir um objetivo nas fases do jogo. Os comandos que o jogador pode utilizar para a construção do algoritmo são simples e devidamente introduzidos no conceito do gameplay do jogo, para que o jogador possa compreender o propósito dos mesmos no objetivo principal do jogo. O *serious game*, ao ser solicitado pelo jogador, move o personagem para demonstrar os passos que foram executados, proporcionando assim o conceito da simulação do algoritmo.

O *serious game* é estruturado de forma a desafiar o jogador ao longo das fases, adicionando níveis de dificuldade conforme o jogador avança. Assim, o mesmo foi dividido em quatro fases. Na primeira fase, o jogador deve utilizar apenas algoritmos sequenciais para passar de fase, tendo apenas uma pilha de comandos para fazê-lo. Na segunda fase, o usuário deve utilizar, além da pilha convencional de comandos, uma estrutura de loop para criar os algoritmos para passar de fase. A partir da terceira fase, é liberado apenas a pilha convencional de comandos e a estrutura de procedimentos, de forma que o usuário possa compreender os conceitos de loop e procedimento de uma forma onde não haja interferência

entre os mesmos. Por fim, a quarta e última fase apresenta todos os conceitos ao usuário, permitindo que ele resolva o objetivo utilizando tanto loops quanto procedimentos, além da pilha de comandos convencional. A pilha de comandos principal tem um número máximo de 12 comandos, exigindo assim que os usuários utilizem, nas fases seguinte à primeira, as estruturas adicionais apresentadas.

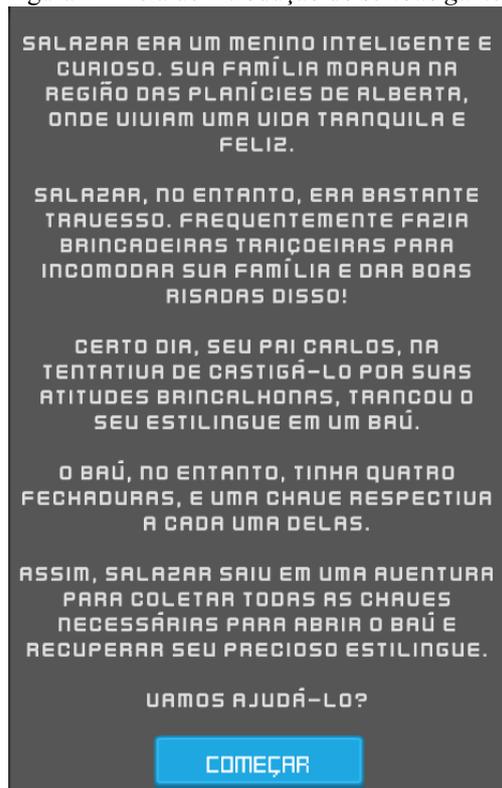
B. Camada da Narrativa

Existem, conforme [5], duas perspectivas de narrativa em jogos: a narrativa do designer e a do jogador. A narrativa do designer é aquela projetada dentro do jogo. Ela pode ser utilizada para preparar o cenário; dar propósito e engajar; prover conteúdo; além de outras utilidades.

A camada de narrativa deste *serious game* baseia-se na história fictícia e original do personagem Salazar. Na introdução do jogo, a seguinte história é apresentada ao usuário, conforme Figura 2.

Os objetivos dos cenários das fases estão estruturados no contexto da história, onde os usuários devem ajudar Salazar a coletar todas as chaves para abrir o baú, onde cada uma delas se encontra no fim de uma fase.

Figura 2 – Tela de introdução do *serious game*.



C. Camada de Gameplay

O *serious game* proposto tem seu gênero definido como um quebra-cabeças (*puzzle*), assim como o Robo Logic e o Lightbot. O jogador deverá estruturar uma pilha de ações (comandos), na estrutura e ordem correta, para mover o personagem até o objetivo do cenário.

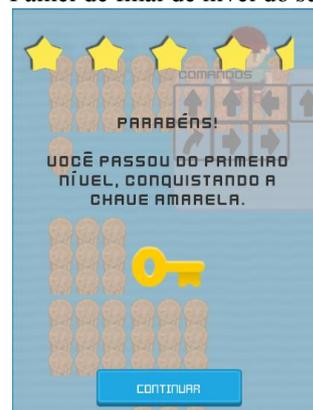
O jogador poderá usar, conforme a Tabela 3, botões de ação (comandos) para estruturar os movimentos do personagem.

Os comandos de repita e procedimento, no entanto, são liberados apenas nas fases nas quais são necessários, de forma que o jogador não seja sobrecarregado pelas opções disponíveis.

Tabela 3 – Botões de ação do jogo.

Botão	Significado do Botão
	Mover-se para Cima
	Mover-se para Baixo
	Mover-se para Esquerda
	Mover-se para Direita
	Pular
	Acionar o Repita
	Acionar o Procedimento
	Play – Executar os comandos

Figura 3 – Pannel de final de nível do *serious game*.



Ao finalizar a estrutura das ações que devem ser executadas, o jogador deve pressionar o botão “play”, conforme Tabela 3, na tela para que a simulação da execução da pilha de comandos seja executada, movendo o personagem no cenário. Caso o usuário tenha executado movimentos errados, a mecânica do jogo procura demonstrar isso, tentando mover o personagem no movimento proposto, mas voltando à posição anterior. Sempre que o usuário efetua um movimento errado, é somado este erro, de forma que seja mostrado no fim da fase, conforme Figura 3, quantas estrelas o usuário recebeu naquele nível. Caso o usuário não realize nenhum movimento errado, será computado uma nota de cinco estrelas. A cada movimento errado, o usuário perderá meia estrela. Assim, com dez erros ou mais, o usuário não recebe nenhuma estrela de nota. Esta implementação tem o objetivo de prover ao usuário uma experiência de feedback de performance e um senso de competitividade ao *serious game*.

Ao carregar as fases, são abertos alguns painéis de instruções que dão dicas da jogabilidade ao jogador e explicam como utilizar os botões da tela. Na segunda e terceira fase são mostradas, respectivamente, algumas dicas sucintas de como o repita e o procedimento são utilizados.

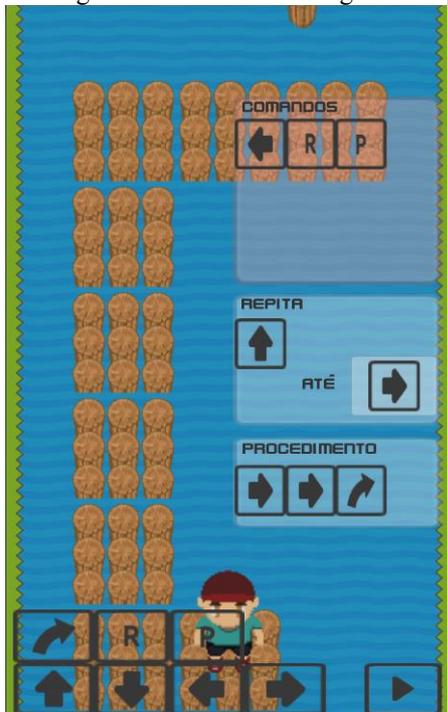
D. Camada de Experiência do Usuário

A camada de experiência do usuário construída neste protótipo utiliza uma perspectiva Top-Down, abordada em um ambiente 2D.

A construção das texturas do cenário foi idealizada com a intenção de mantê-lo contextualizado na narrativa proposta, ao mesmo tempo em que se procura torná-lo divertido e interessante ao público-alvo do trabalho. As tiles utilizadas na criação do cenário foram todas desenvolvidas, criando uma ponte na qual o personagem, Salazar, caminha para atingir seus objetivos em cada cenário. As texturas de fundo, apesar de adquiridas na Internet, tiveram algumas modificações pontuais para se adequarem ao contexto deste *serious game*. A animação da água, por sua vez, foi criada para melhorar o visual e a experiência do jogo, e consiste em *sprites* diferentes, onde as “ondas” da água mudam de posição.

O *asset* do personagem do jogo foi adquirido na Internet, juntamente com suas animações de caminhar. As animações de pulo, no entanto, foram todas feitas sob demanda para o jogo, de forma que a ação do pulo não parecesse estranha sem uma animação adequada. A Figura 4 apresenta a interface de jogo.

Figura 4 – Tela do serious game.



A trilha sonora de um jogo é um elemento importante de design para torná-lo divertido. Por isso, foram adicionadas duas músicas, com um estilo de RPG, que são selecionadas de forma aleatória em cada uma das fases do jogo.

IV. VALIDAÇÃO E RESULTADOS

A etapa de *playtesting* teve o intuito de validar o protótipo proposto e avaliar os resultados do mesmo. O *playtesting* é, conforme o framework DPE, indispensável durante o desenvolvimento de um *serious game*. Em um processo ideal, o *playtesting* deveria ocorrer várias vezes durante o desenvolvimento do projeto, uma vez que é durante esta fase que encontramos problemas e as principais melhorias para o jogo. Apenas monitorando a jogabilidade a partir do público-alvo podemos compreender o que, de fato, precisa ser ajustado ou melhorado. Como este projeto trata-se de um protótipo, apenas duas sessões de *playtesting* foram realizadas.

A primeira etapa foi aplicada com cinco crianças de 7 anos de idade, e uma de 9 anos. Desta amostra, 6 eram meninos e 1 era menina. Os testes ocorreram num período de 5 dias, em uma escola particular da cidade de Caxias do Sul, durante a aula de programação de jogos. As sessões de teste duraram, em média, 15 minutos para cada usuário. Por participarem de uma aula de programação, estes usuários já tinham uma introdução ao pensamento computacional.

A segunda etapa foi aplicada com cinco crianças, de diferentes escolas, duas do sexo feminino e três do sexo masculino. As crianças tinham 7, 8, 9, 11 e 12 anos, sendo as respondentes de 8 e 11 anos, meninas. Esta etapa num período de 4 dias, fora do ambiente escolar. Além disso, estes testes também duraram uma média de 15 minutos para cada usuário. Esta segunda etapa diferencia-se da primeira etapa pois os usuários não tinham nenhuma introdução básica ao pensamento computacional.

Após testarem o jogo, os usuários responderam algumas perguntas, em forma de entrevista. O questionário foi construído procurando avaliar, principalmente, se os usuários compreendem o objetivo e as teorias apresentadas pelo jogo e se a narrativa e a estética do mesmo são engajadoras. Ou seja, se o jogo é bem-sucedido naquilo que é proposto: ser um *serious game* divertido para os usuários do público-alvo.

É importante ressaltar que, início dos testes, não foi dado aos jogadores nenhuma explicação prévia em relação ao jogo ou o que era esperado dos testes, pois o intuito era avaliar se o jogo era autoexplicativo e se os usuários conseguiriam compreender, por conta própria, como a jogabilidade e o aprendizado do mesmo eram desenvolvidos.

Durante o processo de *playtesting*, foram selecionados aleatoriamente um respondente de 6 anos e cinco respondentes de 7 anos, visto que estes pertencem à faixa-etária do público-alvo original do projeto. No entanto, para melhor medir os resultados, foram entrevistados também um usuário de 8 anos, dois usuários de 9 anos, um usuário de 11 anos e um usuário de 12 anos, totalizando assim os 11 participantes do *playtesting*. Além disso, dos 11 respondentes totais, três deles são do sexo feminino.

Os testes foram realizados procurando dividir os usuários entre aqueles que já tinham algum conhecimento introdutório ao pensamento computacional e aqueles que não tinham. Assim, dos 11 respondentes, seis tinham introdução ao pensamento computacional.

O questionário respondido incluía as seguintes questões:

- Idade
- Sexo

- Você tem algum conhecimento prévio de pensamento computacional?
- O objetivo das fases do jogo é claro?
- Avalie a facilidade de compreensão da jogabilidade do jogo para passar das fases (escala Likert de 1 a 5)
- Avalie a facilidade de compreensão do uso de cada botão de comando (escala Likert de 1 a 5)
- Você compreendeu a função do *loop*?
- Você compreendeu a função do procedimento?
- Avalie a estética da interface do jogo (escala Likert de 1 a 5)
- Avalie a qualidade das músicas do jogo (escala Likert de 1 a 5)
- Você achou o jogo desafiador?
- Você gostou de Salazar?
- Você continuaria jogando se houvessem mais fases?
- Os painéis de instruções são esclarecedores e auxiliam na compreensão do jogo?
- Você achou confortável jogar este jogo em um dispositivo móvel?
- Você tem alguma sugestão de melhoria para o jogo?

Com a conclusão da etapa de *playtesting*, foi possível realizar uma análise mais detalhada dos resultados do protótipo do *serious game* proposto e implementado, interpretando e observando seu uso a partir de usuários do público-alvo do projeto.

Foi possível também, compreender melhor a importância da fase de *playtesting* para o projeto como um todo, visto que é muito difícil prever qual será a percepção do público-alvo. Tendo isso em mente, acredita-se que os resultados obtidos durante o *playtesting* foram bastante positivos. O projeto mostrou, apesar de ser apenas um protótipo, grande potencial como artefato de aprendizagem e diversão. O *serious game* também provou-se desafiador para o público-alvo, o que apontou oportunidades para melhorias.

Ficou claro que a inclusão do loop como conceito de aprendizagem aumentou consideravelmente a complexidade do jogo. Desta forma, foi constatado que o loop causa um aumento inevitável na faixa-etária proposta pelo *serious game*, uma vez que as crianças submetidas aos testes com uma idade de nove anos ou mais conseguiram compreender o conceito de forma um pouco menos custosa.

Outro fator importante, que impacta na complexidade do jogo, são os painéis de instruções. Apesar dos usuários terem passado um resultado positivo à esta questão no durante a entrevista, os painéis de instruções certamente poderiam ter uma implementação mais trabalhada, pois houve casos onde alguma interferência externa foi necessária, de forma a auxiliar os usuários na compreensão das teorias apresentadas. Estes painéis de instruções também poderiam ser escritos com uma linguagem mais aderente à faixa-etária proposta pelo projeto.

Ao observarmos os usuários interagindo com o *serious game*, é possível concluirmos também que, o texto de

narrativa apresentado na introdução do jogo é ineficiente e pouco atraente à faixa-etária do público respondente a pesquisa. Apesar de alguns usuários terem lido a narrativa (somente as crianças de 11 e 12 anos), muitos deles não o fizeram pois há muito texto, o que não é apelativo aos mesmos. A apresentação da narrativa do jogo seria muito mais bem-sucedida se a mesma fosse animada.

Observamos também que, como a implementação inicial do protótipo contava com um botão para excluir os comandos das pilhas, isso tornava esta ação confusa aos usuários, pois os mesmos não conseguiam compreender como realizá-la. Assim, o jogo foi alterado para permitir apagar comandos das pilhas de comandos clicando diretamente na imagem dos mesmos, ao invés de utilizar um botão. Além de proporcionar uma melhoria estética, isso permitiu que o usuário excluísse comandos do meio da pilha, e não apenas o último, criando uma jogabilidade mais intuitiva, agradável e fluída.

Outra análise que podemos fazer em relação à implementação deste protótipo é que, na sua concepção, os comandos errados eram apresentados ao usuário com um movimento “em falso” do personagem, onde o mesmo realizava uma tentativa de mover-se na direção proposta e volta à posição anterior. Ao realizar os testes com os usuários da faixa-etária proposta, no entanto, foi possível perceber que esta ação não fica tão evidente aos mesmos quanto se desejava. Uma solução para esta situação seria adicionar um som que indicasse um comando errado durante a movimentação do personagem. Outra solução mais complexa, e provavelmente mais eficiente, seria fazer com que o personagem Salazar caísse na água quando houvessem movimentos errados, retornando o mesmo ao início da fase. Estas alternativas poderão ser consideradas em trabalhos futuros.

V. CONSIDERAÇÕES FINAIS

A partir da investigação dos conceitos e teorias do pensamento computacional, foi possível compreender melhor e com mais clareza como pode ser desenvolvido. Podemos, também, concluir que o pensamento computacional é útil a todos, independentemente da área.

Durante o desenvolvimento do trabalho, investigou-se também os conceitos dos *serious games* e as características que os tornam distintos de jogos comerciais convencionais. Compreendeu-se que um *serious game* deve, acima de tudo, manter-se focado em sua tarefa de prover conhecimento ao jogador, ao mesmo tempo que procura tornar a aprendizagem dos conceitos divertida e engajante. Buscou-se, também, elencar alguns trabalhos relacionados, cujo objetivo de ensino assemelha-se a este, para compreender como são estruturados.

A motivação para criar o protótipo do *serious game* para plataformas *mobile* se deve ao fato de que os jovens estão atualmente muito conectados a estas plataformas.

A pesquisa do framework DPE (Design, Play, Experience) permitiu encontrar uma forma apropriada de modelar o *serious game*, dividindo-o em camadas e enfocando a experiência do jogador. Compreendeu-se, também, a complexidade de um *serious game*, uma vez que o projeto passou pelas fases de design, de prototipação e de testes.

Vivenciar, observar e poder participar de uma fase de *playtesting*, principalmente realizada com crianças, é sem dúvidas uma experiência única. Não é possível prever todos os comportamentos das crianças diante do jogo. Após a aplicação dos testes com as crianças, ficou evidente que o conceito de estrutura de repetição incluído no jogo situa-se no nível de desenvolvimento Operacional Formal, ou, como Piaget indica, na etapa que inicia aos 11 anos de idade, em que é desenvolvida a habilidade de pensar em conceitos abstratos e testar hipóteses logicamente.

AGRADECIMENTOS

Os autores agradecem aos organizadores do X SECIMSEG pelo espaço de discussão e reflexão e aos professores do PPGECiMa pelas sugestões na redação do artigo.

VI. BIBLIOGRAFIA

- [1] Laamarti, F.; Eid, M.; El Saddik, A. An Overview of Serious Games. International Journal of Computer Games Technology, 2014.
- [2] Fardo, M. L. A gamificação como estratégia pedagógica: Estudo dos Elementos dos Games aplicados em processos de ensino e aprendizagem. Universidade de Caxias do Sul. Caxias do Sul. 2013.
- [3] Wing, J. M. Computational Thinking. Center for Computational Thinking, Pittsburgh, Março 2006. Disponível em: <<http://www.cs.cmu.edu/~CompThink/papers/Wing06.pdf>>. Acesso em: Outubro de 2022.
- [4] Jonassen, D. H. Learning to Solve Problems - An Instructional Design Guide. San Francisco: Pfeiffer, 2004.
- [5] Wing, J. Computational thinking and thinking about computing. Center for Computational Thinking, 2008. Disponível em: <<https://www.cs.cmu.edu/~CompThink/papers/Wing08a.pdf>>. Acesso em: Outubro de 2022.
- [6] Mcleod, S. Jean Piaget. SimplyPsychology, 2015. Disponível em: <www.simplypsychology.org/piaget.html>. Acesso em: Outubro de 2022.
- [7] Vygotsky, L. S. A Formação Social da Mente: O Desenvolvimento dos Processos Psicológicos Superiores. 6ª. ed. São Paulo: Martins Fontes, 1998.
- [8] C. Abt, C. Serious Games. Lanham: University Press of America, 1987.
- [9] Costa, S. S. et al. Um estudo exploratório dos games para introdução ao pensamento computacional. 7º Congresso Nacional de Ambientes Hiperídia para Aprendizagem, São Luis, 2015.
- [10] Winn, B. M. The Design, Play, and Experience Framework. Handbook of Research on Effective Electronic Gaming in Education, Michigan State University, v. III, p. 1010-1024, 2008. Disponível em: <http://gel.msu.edu/winn/Winn_DPE_chapter_final.pdf>. Acesso em: Outubro de 2022.
- [11] Anderson, L. W. e Krathwohl, D. R., et al (Eds.) "A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives." Allyn & Bacon. Boston, MA (Pearson Education Group), 2001.