

UNIVERSIDADE DE CAXIAS DO SUL
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
ÁREA DO CONHECIMENTO DE HUMANIDADES
PROGRAMA DE PÓS-GRADUAÇÃO EM EDUCAÇÃO
CURSO DE MESTRADO

Luís Filipe Severgnini

SERIOUS GAMES E O DESENVOLVIMENTO DO PENSAMENTO
COMPUTACIONAL: UMA ABORDAGEM VIGOTSKIANA

CAXIAS DO SUL
2020

Luís Filipe Severgnini

***SERIOUS GAMES* E O DESENVOLVIMENTO DO PENSAMENTO
COMPUTACIONAL: UMA ABORDAGEM VIGOTSKIANA**

Dissertação apresentada à Banca Examinadora do Programa de Pós-Graduação em Educação da Universidade de Caxias do Sul, como requisito para obtenção do título de Mestre em Educação.

Linha de pesquisa: Educação, Linguagem e Tecnologia.

Orientadora: Profa. Dra. Eliana Maria do Sacramento Soares.

CAXIAS DO SUL

2020

Dados Internacionais de Catalogação na Publicação (CIP)
Universidade de Caxias do Sul
Sistema de Bibliotecas UCS - Processamento Técnico

S498s Severgnini, Luís Filipe
Serious games e o desenvolvimento do pensamento
computacional : uma abordagem vigotskiana / Luís Filipe Severgnini. –
2020 73 f. : il. ; 30 cm

Dissertação (Mestrado) - Universidade de Caxias do Sul, Programa
de Pós-Graduação em Educação, 2020.

Orientação: Eliana Maria do Sacramento Soares.

1. Educação. 2. Jogos eletrônicos. 3. Aprendizagem. 4. Ensino
auxiliado por computador. 5. Vigotsky, Lev Semenovich - 1896-1934. I.
Soares, Eliana Maria do Sacramento, orient. II. Título.

CDU 2. ed.: 37

Catalogação na fonte elaborada pela(o) bibliotecária(o)
Carolina Machado Quadros - CRB 10/2236

“Serious games e o desenvolvimento do pensamento computacional: uma abordagem vigotskiana”

Luís Filipe Severgnini

Dissertação de Mestrado submetida à Banca Examinadora designada pela Coordenação do Programa de Pós-Graduação em Educação da Universidade de Caxias do Sul, como parte dos requisitos necessários para a obtenção do título de Mestre em Educação. Linha de Pesquisa: Educação, Linguagem e Tecnologia.

Caxias do Sul, 17 de março de 2020.

Banca Examinadora:

Dra. Eliana Maria do Sacramento Soares (presidente – UCS)

Dra. Elisa Boff (UCS)

Dr. Francisco Catelli (UCS)

Participação por videoconferência

Dr. Eliseo Reategui (UFRGS)

*Dedico este trabalho aos meus pais, que me ensinaram a perseverar,
mesmo nos momentos mais difíceis.*

Agradecimentos

Há tempos, uma certa pessoa me disse que era importante agradecer, pois nada que alcançamos nesta vida o fazemos verdadeiramente sozinhos. Naquele momento, não dei muita importância para isso. Porém, hoje, ao terminar esta dissertação, penso de outra forma: durante os dois anos de jornada, sempre houve alguém ao meu lado, participando da minha caminhada e me ajudando, fosse com uma ideia, uma crítica ou uma piada, fosse com um abraço, um chimarrão ou uma xícara de café. É meu dever agradecê-las.

Primeiramente, gostaria de agradecer aos meus pais, Liane e Luís Carlos, pela sabedoria, pela paciência, pela educação, pelo carinho. Eu não teria chegado até aqui sem o seu apoio.

Em segundo lugar, agradeço à minha namorada, Bárbara, pelo companheirismo, pelo carinho, pelo amor, pela paciência e por ser um porto seguro em que eu pude buscar conforto nos momentos de tormenta.

Agradeço aos professores do mestrado, principalmente à minha orientadora, Profa. Eliana, pelas oportunidades de aprendizagem, pela compreensão e por mostrar novos horizontes, tanto intelectualmente quanto na forma de enxergar a vida.

Agradeço aos meus colegas de mestrado, em especial às minhas colegas e parceiras de orientação, Carol e Vivi, pelas trocas de experiências, por estarem lá nos momentos em que pensávamos em desistir, quando precisávamos de alguém para desabafar ou só queríamos reclamar de alguma coisa do mestrado mesmo. Também mando um salve em especial pro Grupo do Grupo (Ursal).

Agradeço aos meus amigos, pela compreensão e por permanecerem me apoiando, mesmo estando ausente.

Agradeço à escola Zelinda Rodolfo Pessin, escola em que estudei em meus tempos de piá, lugar em que fiz grandes amizades e que sempre me recebeu de portas abertas. Agradeço sinceramente aos alunos que se dispuseram a acordar cedo aos sábados pela manhã para participar da pesquisa. Agradeço também à diretora Jaqueline, pelos incentivos e por estar comigo durante toda a oficina.

Por fim, agradeço a todos aqueles que participaram de alguma forma na construção deste trabalho.

Muito obrigado! Grazie! Muchas gracias! Thank you! Merci! Danke! Arigatō!

*Dizem, com razão, que todo edifício é construído
pedra por pedra, e o mesmo pode ser dito sobre o
conhecimento, extraído e compilado por muitos homens
eruditos, cada um dos quais se baseando no trabalho
daqueles que o precederam. O que um não sabe é
sabido por outro, e pouco permanece verdadeiramente
desconhecido para quem procura com empenho.
Agora eu, Mestre Yandel, assumo o posto de pedreiro,
entalhando o que sei para colocar mais uma pedra
no grande baluarte do conhecimento que vem sendo
construído ao longo dos séculos dentro e fora dos
limites da Cidadela – um baluarte erguido por
incontáveis mãos que viveram antes e que, sem
dúvida, continuará a crescer com a ajuda
de incontáveis mãos que ainda virão.
(O mundo de gelo e fogo, contracapa)*

Resumo

O pensamento computacional emerge no atual contexto sócio-histórico cibercultural como uma forma de pensamento que oferece boas perspectivas, tanto no âmbito da programação quanto no de outras áreas de conhecimento. Ao mesmo tempo, pesquisas relacionadas a jogos digitais apontam que esse tipo de mídia tem grande potencial para o ensino e a aprendizagem. Levando isso em consideração, o problema desta pesquisa consistiu em investigar de que formas os *serious games* contribuem para o desenvolvimento do pensamento computacional em alunos dos Anos Finais do Ensino Fundamental. Nesse sentido, esta pesquisa teve como objetivo geral identificar as contribuições de jogos desse gênero no desenvolvimento de tal forma de pensamento e elaborar uma proposta de prática pedagógica que utilizasse esse tipo de jogo como elemento mediador. Para alcançar tal objetivo, primeiramente, construímos um referencial teórico articulando os conceitos de educação, *serious games* e pensamento computacional, com a abordagem *vigotskiana* de aprendizagem e desenvolvimento. Em seguida, elaboramos os norteadores teóricos que consolidaram a fundamentação teórica da pesquisa. Para conduzir a pesquisa, escolhemos como método o estudo de caso, uma abordagem qualitativa. O caso estudado foi uma oficina de Introdução à Ciência da Computação, mediatizada por um *serious game*, realizada em uma escola pública, composta por sujeitos dos Anos Finais do Ensino Fundamental. O corpus da pesquisa foi constituído pelas produções dos alunos no *serious game* e pela documentação das atividades por meio de anotações e de filmagens. Dentre as atividades realizadas na oficina, destacamos a exploração, a exposição dialogada, a resolução de problemas e a programação em pares. A análise do corpus foi realizada a partir de dois procedimentos distintos, ambos fundamentados nos norteadores teóricos: os dados textuais (anotações e transcrições das filmagens) foram analisados por meio de análise textual discursiva, enquanto as soluções dos alunos no jogo (programas de computador) foram analisadas por uma técnica autoral, que examinou qualitativamente quais conceitos do pensamento computacional foram aprendidos em quais níveis do jogo e por quais alunos. Os resultados da análise apontam que houve indícios de internalização de conceitos do pensamento computacional no caso estudado, tanto em termos qualitativos quanto quantitativos. Essa internalização foi mediatizada pelo jogo *CodeCombat*, o que indica que os *serious games* contribuem para o desenvolvimento dessa forma de pensamento. Os aspectos relacionados aos *serious games* que possibilitam isso, podem ser categorizados como: mediatização, em que se incluem a interatividade, a possibilidade de aprender com os erros, a cibercultura, a cultura dos jogos e a promoção da sociointeração; desafios, problemas no contexto do jogo que buscam mediar o acesso ao objeto de conhecimento, aos conceitos relacionados ao pensamento computacional *per se*; mediação *vigotskiana*, a capacidade do outro mais experiente de fortalecer a compreensão e a atribuição de sentido do aluno em relação aos desafios do jogo. Os achados da pesquisa também corroboram a noção de que o potencial mediador de um *serious game* pode ser mais bem explorado quando aliado à mediação do professor. Com base nesses resultados, criamos uma proposta de prática pedagógica que utiliza o *serious game CodeCombat* como elemento mediador. Por fim, ressaltamos que a abordagem *vigotskiana* utilizada nesta pesquisa pode ser ampliada para outros contextos, não se limitando aos *serious games*. Propomos que seus resultados possam ser transpostos a ambientes onde existam outros recursos, como objetos de aprendizagem, jogos de entretenimento, ambientes de programação como o *Scratch*, e até mesmo atividades de computação desplugada.

Palavras-chave: Educação. Jogos digitais. Mediação. Vigotski. Pensamento computacional.

Abstract

Computational thinking emerges in the current cybercultural, socio-historical context as a way of thinking that offers good perspectives, both in programming and in other areas of knowledge. At the same time, gaming related research shows that such kind of media has a great potential for teaching and learning. Taking this into account, this research's problem was to investigate how serious games contribute to computational thinking development in students of 7th and 8th grades. In that sense, this research's main goal was identifying the contributions of serious games in the development of such a way of thinking, as well as elaborating a pedagogical practice proposal that involved a game of such genre as a mediating element. To achieve this goal, first, we build a theoretical framework articulating the concepts of education, serious games and computational thinking, with the Vigotskian approach to learning and development. Then, we elaborated the theoretical guidelines that consolidated this research's theoretical foundation. We chose the case study as our method, a qualitative approach, and the case we studied was an Introduction to Computer Science workshop, mediated by a serious game, held in a public school, composed of students from 7th and 8th grades. This research's corpus was constituted by the students' creations inside the serious game and the documentation of activities through notes and filming. Among the activities carried out in the workshop, we highlight: exploration, dialogued exposition, problem solving and pair programming. Analysis was performed by using two different procedures, both grounded upon the theoretical guidelines: textual data (notes and recordings transcription) was analyzed through Textual-Discursive Analysis, while the students' creations inside the game (computer programs) were analyzed with a custom method, which examined qualitatively what computational thinking concepts were learned in what game levels and by which students. Results show evidences of internalization of computational thinking concepts, both in qualitative and quantitative terms. Such internalization was mediatized by CodeCombat, which indicates serious games contribute to computational thinking development. Aspects related to serious games that allow such phenomenon may be categorized as: *mediatization*, in which we include interactivity, the possibility of learning from failure, cyberculture, gaming culture and social interaction promotion; *challenge*, problems in the gaming context that seek to mediate access to the learning subject, concepts related to computational thinking itself; *vigotskian mediation*, the more knowledgeable other's ability to strengthen the student's understanding and sense making in relation to the game challenges. The research findings also corroborate the notion that the mediatization potential of a serious game can be better explored when combined with the mediation of a MKO. Based on results, we created a pedagogical practice proposal that uses CodeCombat as its mediating element. Lastly, we emphasize that this research's Vigotskian approach may be expanded to other contexts, not limiting itself to serious games. We propose that results may be transposed to environments where other learning resources are used, such as learning objects, entertainment games, programming platforms like Scratch, and even unplugged computing activities.

Keywords: Education. Games. Mediation. Vygotsky. Computational thinking.

Lista de ilustrações

Figura 1 – Nuvens de palavras: transcrições dos vídeos	70
Figura 2 – Nuvens de palavras: anotações do pesquisador	70
Figura 3 – Situações de <i>feedback</i> em <i>CodeCombat</i>	76
Figura 4 – Ciclo de Mediatização Interativa	78
Figura 5 – Sociointeração e mediação do professor em um encontro da oficina . . .	85
Figura 6 – <i>Screenshot</i> do Nível 1 - “Masmorra Kithgard”, durante a apresentação dos objetivos	88
Figura 7 – <i>Screenshot</i> do Nível 2 - “Gemas nas Profundezas”	90
Figura 8 – <i>Screenshot</i> do Nível 3 - “Guarda sombrio”	91
Figura 9 – Duas possíveis linhas de raciocínio para resolver o Nível 3 - “Guarda Sombrio”	92
Figura 10 – Mapa dos níveis do primeiro mundo de <i>CodeCombat</i>	93
Figura 11 – <i>Screenshot</i> do Nível 4 - “Mina Inimiga”, primeiro nível a exigir uma solução eficiente	95
Figura 12 – <i>Screenshot</i> do Nível 5 - “Nomes Próprios”, em que o comando de ataque é introduzido	97
Figura 13 – Objetivos do desafio combo “Hora de Dormir”: comandos básicos, argumentos e <i>strings</i>	99
Figura 14 – <i>Screenshot</i> do Desafio Combo “Hora de Dormir”	100
Figura 15 – “O Prisioneiro”: além de pensamento computacional, o nível exige criatividade e muitas tentativas	101
Figura 16 – Algumas das linhas de raciocínio possíveis para resolver o problema d’O Prisioneiro	102
Figura 17 – <i>Screenshot</i> do desafio de conceito “Loop no Armazém”	104
Figura 18 – Reconhecimento de padrões: série de instruções que, quando repetidas, solucionam o “Loop no Armazém”	104
Figura 19 – Momento inicial da animação da trajetória do herói no armazém após a elaboração de uma solução correta	105
Figura 20 – Avaliação dos desafios de conceito e combo – quais conceitos foram usados?	112
Figura 21 – Relações entre as categorias emergentes e os norteadores teóricos da pesquisa	120
Figura 22 – Ambientação do jogo <i>CodeCombat</i> e exemplo de solução (à direita) . .	141
Figura 23 – Objetivos do primeiro nível do jogo <i>CodeCombat</i>	142

Lista de tabelas

Tabela 1 – Processos psicológicos elementares e superiores	43
Tabela 2 – Conceitos relacionados ao pensamento computacional	55
Tabela 3 – Sujeitos da pesquisa	62
Tabela 4 – Norteadores teóricos	67
Tabela 5 – Alunos x Níveis Concluídos	110
Tabela 6 – Conceitos aprendidos pelos alunos na visão do jogo	113
Tabela 7 – Níveis x Conceitos do Pensamento Computacional	114
Tabela 8 – Conceitos do Pensamento Computacional x Alunos	115
Tabela 9 – Classificações das soluções	116
Tabela 10 – Alunos x Níveis Concluídos (com classificação e ponderação)	117
Tabela 11 – Conceitos do Pensamento Computacional x Alunos (com classificação e ponderação)	118

Lista de abreviaturas e siglas

ATD	Análise Textual Discursiva
BNCC	Base Nacional Comum Curricular
CMI	Ciclo de Mediatização Interativa
CSTA	<i>Computer Science Teachers Association</i>
IDEB	Índice de Desenvolvimento da Educação Básica
IFRS	Instituto Federal do Rio Grande do Sul
ISTA	<i>International Society for Technology in Education</i>
MKO	<i>More Knowledgeable Other</i>
PC	Pensamento Computacional
PGB	Pesquisa Games Brasil
PPGEdu	Programa de Pós-Graduação em Educação
ProUni	Programa Universidade para Todos
RPG	<i>Role-playing Game</i>
UCS	Universidade de Caxias do Sul
TDIC	Tecnologias Digitais da Informação e Comunicação
TI	Tecnologia da Informação
ZDP	Zona de Desenvolvimento Proximal

Lista de trechos de código

1	Solução do autor para o nível “Gemas nas profundezas”	89
2	Solução sugerida pelo jogo para o nível “Gemas nas profundezas”	90
3	Solução não-otimizada para o nível “Mina inimiga”	96
4	Solução ótima para o nível “Mina inimiga”	96
5	Solução ótima para o Desafio Combo “Hora de Dormir”	99
6	Solução para o nível 9 - “Dança no Fogo”	103
7	Solução para o desafio de conceito “ <i>Loop</i> no Armazém”	105

Sumário

1	INTRODUÇÃO	31
1.1	Apresentação	31
1.2	Contextualização	32
1.2.1	Tema de pesquisa	34
1.3	Objetivos e questão de pesquisa	35
1.4	Estrutura da dissertação	36
2	REFERENCIAL TEÓRICO	39
2.1	Educação na cultura digital	39
2.2	Aprendizagem à luz da teoria vigotskiana	41
2.2.1	Processos psicológicos superiores	42
2.2.2	Mediação	44
2.2.3	Internalização	45
2.2.4	Aprendizagem	45
2.2.4.1	Relações entre aprendizagem e desenvolvimento	46
2.2.5	Zona de desenvolvimento proximal	48
2.3	<i>Serious games</i>	49
2.3.1	Possibilidades e limitações	50
2.4	Pensamento computacional	53
2.4.1	Pensamento computacional? Por que não matemática ou lógica formal?	57
3	MÉTODO	59
3.1	Procedimentos metodológicos	60
3.1.1	Estudo de caso	61
3.1.2	Descrição do caso	62
3.1.2.1	Estudo piloto	63
3.1.2.2	<i>Corpus</i> de pesquisa	63
3.1.2.3	Atividades realizadas na oficina	64
3.1.2.4	Aspectos éticos da pesquisa	65
3.2	Procedimentos de análise	66
3.2.1	Análise textual discursiva	66
3.2.2	Análise das produções do jogo	71
4	RESULTADOS E DISCUSSÃO	73
4.1	Categorias emergentes	73
4.1.1	Mediatização	73

4.1.1.1	Interatividade: potencializando a elaboração de hipóteses e a resolução de problemas	74
4.1.1.2	Aprendendo com os erros	80
4.1.1.3	Cultura dos jogos digitais e cibercultura	81
4.1.1.4	Relações entre mediatização, sociointeração e internalização	83
4.1.2	Desafio	87
4.1.2.1	Critérios de escolha dos desafios analisados	87
4.1.2.2	Masmorra Kithgard: introdução ao pensamento computacional	89
4.1.2.3	Gemas nas Profundezas: pensamento algorítmico e argumentos	89
4.1.2.4	Guarda Sombrio: pensamento lógico e algorítmico em ação conjunta	91
4.1.2.5	<i>Kounter Kithwise</i> : níveis complementares, padrões e generalização	92
4.1.2.6	Passos Cuidadosos: desafios de conceito como avaliação em <i>CodeCombat</i>	93
4.1.2.7	Mina Inimiga: aprofundando a avaliação da solução de um problema	94
4.1.2.8	Nomes Próprios: atacando, reconhecendo padrões e generalizando	96
4.1.2.9	Hora de Dormir: decomposição em desafios <i>combo</i>	98
4.1.2.10	O Prisioneiro: tentativa e erro, criatividade e pensamento computacional	100
4.1.2.11	<i>Loop</i> no Armazém: reconhecendo padrões (onde parece não haver)	102
4.1.3	Mediação	105
4.1.3.1	Mediação, MKO, ZDP e internalização	106
4.1.3.2	A abordagem <i>vigotskiana</i> do mediador	108
4.2	Análise das produções dos alunos	109
4.2.1	Aprendizagem sob a ótica do jogo <i>CodeCombat</i>	110
4.2.2	Aprofundando a análise	113
4.3	Ligando os pontos: uma possível resposta à questão de pesquisa	119
4.3.1	Como um <i>serious game</i> desenvolve o pensamento computacional?	119
4.3.2	Algumas observações necessárias	122
4.4	Uma proposta de prática pedagógica com <i>serious games</i>	124
5	CONSIDERAÇÕES FINAIS	129
	REFERÊNCIAS	133
	APÊNDICES	139
	APÊNDICE A – O <i>SERIOUS GAME CODECOMBAT</i> EM UMA CASCA DE NOZ	141
	APÊNDICE B – TERMO DE ANUÊNCIA INSTITUCIONAL	145
	APÊNDICE C – TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO	147

APÊNDICE D – TERMO DE ASSENTIMENTO LIVRE E ESCLA- RECIDO	153
APÊNDICE E – PROGRAMA EM PYTHON Nº1	159
APÊNDICE F – NIVEIS_ALUNOS.CSV	163
APÊNDICE G – NIVEIS_CONCEITOS_PC.CSV	165
APÊNDICE H – PROGRAMA EM PYTHON Nº2	167
APÊNDICE I – NIVEIS_CONCEITOS_PC_QUALI.CSV	171
APÊNDICE J – NIVEIS_ALUNOS_QUALI.CSV	173
APÊNDICE K – LEGENDAS_QUALI_AVALIACAO.CSV	175

1 Introdução

1.1 Apresentação

Antes de contextualizar o problema que originou esta pesquisa, penso¹ ser importante falar um pouco sobre mim, para que se possa conhecer as raízes das angústias que me levaram a pesquisar acerca de temas como o pensamento computacional e os jogos digitais.

Iniciei meus estudos em Farroupilha, minha cidade natal. Alguns bons anos depois, já no último semestre do ensino médio, ingressei no curso Técnico em Informática, do IFRS Campus Farroupilha. Ao final desse mesmo ano, ganhei uma bolsa de estudos integral do ProUni (Programa Universidade para Todos), na Universidade de Caxias do Sul (UCS), decorrente de meu desempenho no Exame Nacional do Ensino Médio. A experiência que tive no primeiro semestre do curso técnico direcionou-me à área da computação.

Apesar disso, não sabia com exatidão qual dos cursos era o mais adequado ao meu perfil, tampouco qual corresponderia melhor às minhas expectativas de futuro. Provavelmente, essa incerteza – combinada com o apreço pelos *games* – foi o fator que me fez escolher o curso Bacharelado em Tecnologias Digitais, cuja multidisciplinaridade curricular parecia fornecer uma base consistente para um pretendo desenvolvedor de jogos digitais.

Após ter me adaptado como aluno desse curso, tendo concluído significativa parcela das disciplinas, tive a oportunidade de conhecer outra universidade, a *Dalhousie University*, no Canadá, por meio do programa Ciência sem Fronteiras². Durante meu intercâmbio, estudei Ciência da Computação por dois quadrimestres e cumpri estágio de três meses no departamento de TI da biblioteca da universidade. Ao retornar para o Brasil, já sem a bolsa de estudos, dei continuidade ao curso Bacharelado em Tecnologias Digitais, decidindo que meu trabalho de conclusão envolveria *game design* e ensino de lógica de programação. Assim, projetei e desenvolvi um *serious game* para essa finalidade. No decorrer desse trabalho, tive a oportunidade de me aproximar de algumas teorias da aprendizagem, como a abordagem histórico-cultural de Vigotski. Esse trabalho de conclusão certamente preparou o terreno para meu ingresso na área da educação.

Atualmente, estou próximo da conclusão do curso de Mestrado em Educação, do

¹ Em virtude de ser um momento no qual descrevo minha trajetória escolar, profissional e acadêmica, preferi escrever esta seção em primeira pessoa.

² Programa do governo federal que buscava promover a consolidação, expansão e internacionalização da ciência e tecnologia do país, por meio do intercâmbio e da mobilidade internacional. Alguns dos países parceiros do programa Ciência sem Fronteiras: Canadá, EUA, Austrália, Nova Zelândia, Reino Unido, Coreia do Sul, entre outros.

PPGEdu da UCS – em que me empenhei persistentemente para aprender o máximo possível sobre as relações existentes entre educação, linguagem e tecnologia. Além de uma insaciável vontade de aprender, apresento nesta pesquisa o olhar de alguém que trabalha há pelo menos 7 anos na área de desenvolvimento de *software* e que continua motivado a compreender o que seja o pensamento computacional, como ele se desenvolve e de que forma os jogos digitais podem contribuir para esse desenvolvimento, sob uma perspectiva *vigotskiana* de aprendizagem.

1.2 Contextualização

Durante minha trajetória como acadêmico do curso de Bacharelado em Tecnologias Digitais – que abrange as áreas da arte, comunicação e computação – pude perceber que uma porcentagem considerável dos alunos enfrentava dificuldades nas disciplinas de Algoritmos e Lógica de Programação, disciplinas fundamentais da área da Ciência da Computação. Ainda hoje, em decorrência dessas dificuldades, muitos alunos desistem de seus cursos, o que é evidenciado nos censos do ensino superior brasileiro; na Sinopse Estatística da Educação Superior 2016 (INEP, 2017), por exemplo, as áreas de Ciência da Computação e Processamento da Informação apresentam percentuais acumulados de desistência³ de 33,7% e 40%, respectivamente.

Essa perceptível dificuldade de aprendizagem de algoritmos e lógica de programação sempre despertou minha curiosidade. Por um longo período, uma certa mistura de ingenuidade e arrogância fez-me pensar que fosse normal que as pessoas tivessem dificuldade ao estudar tais disciplinas, como se dominá-las fosse exclusividade para mentes intelectualmente mais capazes. Afinal, por mais injusto que parecesse, minha experiência como aluno e programador, inspirada pelo comportamento de meus professores e de profissionais mais experientes, somada à natureza difícil da Ciência da Computação, sustentava essa percepção.

No entanto, novas impressões demonstraram-me que isso não era necessariamente verdade. Uma dessas impressões foi estritamente pessoal, envolvendo alguns colegas da faculdade. Nos primeiros semestres da graduação, esses colegas tinham o hábito de exteriorizar sua insatisfação com as disciplinas de algoritmos e programação, reclamando principalmente de suas baixas notas e de suas dificuldades de aprendizado. Porém, pouco tempo depois, alguns deles tiveram a oportunidade de trabalhar como estagiários em empresas de desenvolvimento de *software* e de *websites*. A partir disso, eles tiveram uma perceptível mudança quanto ao apreço pela programação: estavam mais satisfeitos com seus programas e passaram a obter melhor desempenho nas disciplinas. Presenciar essa

³ O percentual indicado se refere única e exclusivamente às matrículas desvinculadas. Tal percentual é ainda mais significativo se forem consideradas as matrículas trancadas e transferências para outros cursos.

transformação me fez ponderar as certezas que tivera no passado. Possivelmente, um dos principais motivos pelos quais não conseguiam aprender era a ausência de experiências anteriores nessas disciplinas, não de capacidades.

Outra impressão que tive foi percebida por meio da internet, em uma leitura sobre um projeto chamado *Hour of Code*⁴ (Hora do Código, em português). Nesse projeto, crianças a partir dos quatro anos de idade são apresentadas à programação de computadores e dão seus primeiros passos como desenvolvedores de jogos digitais. Descobrir isso me aproximou da noção de que a experiência prévia num determinado assunto pode ser mais determinante do que uma possível predisposição natural, concepção que vai ao encontro da perspectiva *vigotskiana* de desenvolvimento psicológico (VIGOTSKI, 2007), que pude conhecer ao longo do meu trabalho de conclusão da graduação.

Nessa perspectiva, o desenvolvimento de capacidades como dedução, compreensão, domínio das formas lógicas de pensamento e da lógica abstrata não é apenas um processo biológico: ele é desencadeado pela aprendizagem, que se dá pela interação do indivíduo com o meio sociocultural. Outro aspecto notável dessa abordagem é a ideia de que o processo de aprendizagem das crianças inicia muito antes de elas ingressarem na escola e que esse conhecimento prévio influencia suas trajetórias escolares (VIGOTSKI, 2007), o que significa que cada indivíduo tem facilidades e dificuldades em situações e em graus diferentes. Esse ponto de vista, baseado nas ideias *vigotskianas*, permitiu-me elaborar duas conjecturas acerca do problema investigado:

1. se a aprendizagem ocorre principalmente na interação com o meio sociocultural, e se em ambientes de aprendizagem, como a sala de aula, a organização desse meio é responsabilidade do educador, então a organização do ambiente pelo educador influencia diretamente no processo de aprendizagem dos alunos; logo, a não-aprendizagem por parte dos alunos de Algoritmos e Lógica de Programação poderia indicar uma inadequação na organização do ambiente de aprendizagem, isto é, nos métodos de ensino, nos planos de aula e nas práticas pedagógicas do educador;
2. os alunos ingressam despreparados nos cursos de computação e afins, o que revela um deficit de aprendizado nessa área em seus percursos escolares.

No que diz respeito à primeira conjectura, parece-me razoável sugerir que, independentemente do método de ensino, se o aluno não estiver disposto a aprender, não haverá aprendizagem. Ao mesmo tempo, é possível que a própria inadequação e desatualização dos métodos de ensino de computação esteja relacionada aos elevados índices de reprovação e desistência dos cursos da área, bem como à percepção de não-aprendizagem por parte dos alunos. É importante destacar que entendo como inadequados e desatualizados os

⁴ Fonte: <https://hourofcode.com/pt/pt>

ambientes de ensino em que o aluno é percebido como alguém passivo, um mero receptor do conhecimento. Dessa forma, parece oportuno inovar no sentido de potencializar as chances de mobilizar os alunos, motivando-os a assumir papéis de protagonistas em seus processos de aprendizagem. A segunda conjectura, por sua vez, refere-se ao despreparo dos alunos recém-ingressados nos cursos de computação. Tomando como base a perspectiva de Vigotski, entendo que

Onde o meio não cria os problemas correspondentes, não apresenta novas exigências, não motiva nem estimula com novos objetivos o desenvolvimento do intelecto, o pensamento do adolescente não desenvolve todas as possibilidades que efetivamente contém, não atinge as formas superiores ou chega a elas com um extremo atraso [sic]. (VIGOTSKI, 2009, p. 171)

Esse excerto corrobora a ideia de que o contato antecipado com problemas relacionados à programação de computadores pode influenciar diretamente no desenvolvimento das competências e habilidades inerentes a essa atividade. Daí a intenção de pesquisar o desenvolvimento do pensamento computacional em pré-adolescentes, em vez de alunos do ensino superior. Nessa linha de raciocínio, jovens que convivem num contexto sociocultural que promove o desenvolvimento do pensamento computacional desde cedo tendem a enfrentar menores dificuldades quando ingressarem no ensino superior.

1.2.1 Tema de pesquisa

O pensamento computacional (PC) é um assunto que tem recebido bastante atenção nos currículos do ensino básico de países como Estados Unidos, Itália, Finlândia e, mais recentemente, Brasil (VALENTE, 2016), principalmente por oferecer boas perspectivas, tanto no âmbito do ensino e aprendizagem de Algoritmos e Lógica de Programação, quanto no de outras áreas do conhecimento. No contexto educacional brasileiro, há menção a esse modo de pensar na Base Nacional Comum Curricular (BNCC), documento que serve como referência para a elaboração dos currículos escolares para o ensino infantil, fundamental e médio (BRASIL, 2017).

Tal modo de pensamento se caracteriza como uma forma de raciocínio que emprega conceitos e técnicas da ciência da computação, como abstração, pensamento algorítmico, reconhecimento de padrões e decomposição, para a resolução de problemas em diferentes contextos (WING, 2006); caracteriza-se como uma habilidade essencial para todos, não somente para cientistas da computação. Considerando sua eminente relevância e uma provável popularização em um futuro próximo, no qual as tecnologias digitais devem permear ainda mais a vida dos indivíduos, o pensamento computacional apresenta-se como um tema de pesquisa atual e promissor.

Nesse sentido, parece razoável aproximarmos esse tema de outro também muito atual e tecnológico: os jogos digitais, que, atualmente, fazem parte da linguagem e da

experiência de mundo de milhões de pessoas, sejam elas mulheres ou homens, crianças ou adultos. Estima-se que 75,5% dos brasileiros joguem jogos eletrônicos (PGB, 2018).

Os jogos digitais (ou *games*) podem servir como motivação extra para os alunos (GEE, 2003; GEE, 2005; PRENSKY, 2003), mas essa não é a única razão para utilizar *games* em ambientes de ensino. Muito além do fator motivacional *per se*, Gee (2003) acredita no potencial educativo dos jogos, particularmente por causa das possibilidades de aprendizagem que eles oferecem, uma vez que apresentam quantidades equilibradas de desafio, ajuda e *feedback*, recompensando o domínio por meio de novos desafios. Entretanto, cabe uma ressalva: para estar vinculado às práticas pedagógicas, um jogo não deve conter elementos impróprios ou negativos, como a representação de violência contra indefesos, o genocídio e a glamorização do consumo de drogas, por exemplo. Por conta disso, existem os *serious games*, jogos nos quais a aprendizagem é o objetivo principal, ao invés do puro entretenimento (MICHAEL; CHEN, 2005), isto é, jogos eletrônicos desenvolvidos com o intuito primário de ensinar, treinar e promover hábitos saudáveis e mudança social (RITTERFELD; CODY; VORDERER, 2009), sem prescindir de elementos como motivação e diversão.

1.3 Objetivos e questão de pesquisa

O objetivo geral desta pesquisa foi identificar as contribuições dos *serious games* para o desenvolvimento do pensamento computacional e, a partir disso, elaborar uma proposta de prática pedagógica em que jogos desse gênero fossem elementos mediadores da aprendizagem de tal forma de pensamento, sob uma perspectiva *vigotskiana*.

Além disso, esta pesquisa também compreendeu os seguintes objetivos específicos:

- construir um quadro teórico, a partir da articulação dos conceitos de educação, aprendizagem, pensamento computacional e *serious games*;
- gerar norteadores a partir das articulações teóricas construídas, com o intuito de possibilitar a construção e a análise do *corpus* de pesquisa;
- constituir um *corpus* de pesquisa a partir dos dados gerados em uma oficina de programação, na qual o *serious game CodeCombat* fosse utilizado como elemento mediador, levando-se em consideração os norteadores teóricos gerados;
- analisar o *corpus* de pesquisa a partir dos norteadores teóricos gerados, buscando identificar as contribuições advindas do jogo *CodeCombat* no processo de desenvolvimento do pensamento computacional.

Para conduzir esta investigação e alcançar os objetivos propostos, considerando-se o contexto e o tema escolhidos, formulamos a seguinte questão de pesquisa: de que forma

os *serious games* podem contribuir para o desenvolvimento do pensamento computacional em alunos dos anos finais do Ensino Fundamental, a partir da perspectiva *vigotskiana*?

1.4 Estrutura da dissertação

Devido a uma possível falta de ousadia, somada ao pouco tempo disponível para a escrita, o formato desta dissertação segue um padrão costumeiramente adotado em trabalhos do gênero: introdução, referencial teórico, método, resultados, discussão e considerações finais. A introdução, que se encerra neste comentário sobre a estrutura da dissertação, apresentou o autor e contextualizou o tema e a questão de pesquisa.

O referencial teórico, capítulo subsequente, objetiva aproximar o leitor do autor no que se refere ao sentido dos conceitos utilizados – é um esforço necessário para que haja uma melhor compreensão entre as partes. Isso posto, o primeiro grande conceito abordado é também possivelmente o mais ambíguo deles: educação. Depois disso, apresentaremos um entendimento sobre a espinha dorsal deste trabalho: a abordagem *vigotskiana* de aprendizagem e desenvolvimento. Por fim, são apresentados os conceitos de *serious games* e de pensamento computacional, temas da pesquisa. A partir dessas definições, espera-se que o leitor possa entender as decisões do autor acerca do método e sua interpretação dos resultados.

O terceiro capítulo narra como a pesquisa foi planejada e executada. Para tanto, explicaremos inicialmente as escolhas da abordagem qualitativa e do estudo de caso como método. Na sequência, o caso estudado: uma oficina de Introdução à Ciência da Computação, mediatizada por um *serious game*, realizada em uma escola pública, composta por sujeitos dos Anos Finais do Ensino Fundamental. Posteriormente, para preparar o terreno da apresentação e da discussão dos resultados, são explicados os procedimentos de análise.

O quarto capítulo é provavelmente o mais significativo desta dissertação, pois retrata uma síntese de todas as vivências ocorridas ao longo dos dois anos de pesquisa. Nele são apresentados e discutidos os resultados da Análise Textual Discursiva e da análise das produções dos alunos, tendo como base os norteadores construídos a partir do referencial teórico. Após a discussão, a fim de condensar tudo que foi aprendido em uma aplicação prática, apresentaremos uma breve proposta de prática pedagógica, elaborada a partir das aprendizagens decorrentes do estudo de caso e dos processos de análise e interpretação do *corpus*.

Finalmente, apresentaremos as considerações finais. Nesse momento, faz-se uma retrospectiva da dissertação como um todo, começando pela recapitulação dos objetivos propostos e avaliando como – ou se – eles foram alcançados. Em seguida, uma breve discussão sobre a contribuição que este trabalho de pesquisa traz à comunidade científica,

bem como quais são suas limitações. Ao final, são apresentadas perspectivas para os próximos passos da pesquisa, no intuito de evidenciar as “pontas soltas” e levantar novas questões norteadoras para aqueles que se interessarem pelo tema.

2 Referencial teórico

2.1 Educação na cultura digital

A educação é um processo perpétuo de constituição, transformação e conservação de saberes, que ocorre por meio das interações entre indivíduos e grupos, mediado pela cultura e pela linguagem. É um processo de construção de conhecimentos, de desenvolvimento de competências, de habilidades e de adoção de atitudes e valores em um meio sociocultural. Dessa definição, podemos entender que o homem é um ser inacabado, que se encontra em constante processo de transformação, ao longo de toda sua vida.

Em consonância com essa concepção, Jayme Paviani (2014) considera a educação uma condição universal, com significado equivalente à vida e à cultura – uma condição que transcende os muros das instituições formais de ensino e permeia todos os lugares, desde as escolas e lares, até os ambientes de trabalho e espaços de lazer.

A educação é um fenômeno econômico, social, político, cultural e histórico; processo de transformação e de conservação de valores, crenças, hábitos, costumes, e ideias dos indivíduos e dos grupos. Educar é mudar e conservar. É renovação e tradição. É uma experiência individual, coletiva e histórica. Por isso o conceito de educação precisa ser buscado continuamente a partir da realidade social e histórica. (PAVIANI, 2014, p. 63).

Concordar com Paviani implica compreender a educação como uma condição humana, um fenômeno que preserva e transforma valores, crenças, hábitos, costumes e ideias, que não se restringe à dimensão do conhecimento. Partilhar dessa definição significa, também, reconhecer que o conceito de educação se modifica ao longo do tempo, a partir da realidade sócio-histórico-cultural.

No atual momento histórico, as tecnologias digitais de informação e comunicação (TDIC) têm transformado a forma como as pessoas vivem e percebem o mundo. Em decorrência da ascensão do ciberespaço¹ e de suas características, ao se relacionar com as TDICs, o ser humano se reconfigurou: ele não precisa mais estar fisicamente presente para interagir com outras pessoas, pois é capaz de se comunicar com elas por meio de aplicações de software de troca de mensagens, textuais ou audiovisuais, tanto de forma

¹ Termo inventado pelo escritor *cyberpunk* William Gibson, autor da obra-prima *Neuromancer*, principal influência da antológica trilogia cinematográfica *Matrix*, para se referir a “um espaço não-físico ou territorial composto por um conjunto de redes de computadores através das quais todas as informações (sob as suas mais diversas formas) circulam” (LEMOS, 2013, p. 127). Levy (2010), por sua vez, entende como ciberespaço não somente o meio de comunicação que surge da interconexão mundial de computadores, mas também o universo de informações nele armazenadas e as pessoas que nele navegam.

síncrona, quanto assíncrona; tampouco precisa se locomover à biblioteca para encontrar as informações que precisa, pois tem acesso a uma parte considerável do acervo de conhecimento humano a poucos cliques de distância.

Essas possibilidades emergem a partir do ciberespaço e caracterizam a cultura digital. Inicialmente concebida por Pierre Lévy como cibercultura, a cultura digital é “o conjunto de técnicas (materiais e intelectuais), de práticas, de atitudes, de modos de pensamento e de valores que se desenvolvem juntamente com o crescimento do ciberespaço” (LEVY, 2010, p. 17). André Lemos (2009, p. 136), por sua vez, define-a como “a cultura contemporânea, onde os diversos dispositivos eletrônicos já fazem parte da nossa realidade”.

Na cultura digital, mudanças tecnológicas ocorrem cada vez mais rapidamente, a ponto de não serem completamente assimiladas pelas pessoas. A velocidade e a forma como as informações são consumidas e disseminadas também se alterou: o acesso à informação ocorre de forma instantânea e o paradigma de comunicação 1:N (um para muitos), típico da cultura de massas, cedeu espaço ao modelo N:N (muitos para muitos), em que os receptores também são emissores, criadores de conteúdo². Como decorrência dessas novas relações com o saber, o conceito de educação também tem se transformado.

Conforme Lyotard (apud KENSKI, 2011), para acompanhar o movimento do mundo, o homem precisa se adaptar à complexidade proposta pela evolução tecnológica. Desta forma, a educação, como processo perpétuo e mutável, não deve se alienar às mudanças causadas por tal evolução. Kenski (2011, p. 18) afirma que isso representa um desafio duplo à educação, principalmente no que se refere à educação escolar, no sentido de que ela precisa se adaptar aos avanços das tecnologias e “orientar o caminho de todos para o domínio e a apropriação crítica desses novos meios”.

Atualmente, as TDICs estão presentes em uma grande parcela das escolas brasileiras. Apesar disso, o IDEB³ observado segue aquém do esperado, principalmente nos Anos Finais do Ensino Fundamental e no Ensino Médio (INEP, 2018). Dessa forma, percebemos que as TDICs não são instrumentos mágicos que garantem a aprendizagem daqueles que as utilizam. Como reitera Kenski (2011), as tecnologias por si próprias não educam ninguém. Moraes (2002) defende que as TDICs estão sendo empregadas apenas como “camuflagem” para metodologias ultrapassadas de ensino, nas quais os alunos são espectadores e reprodutores de informação:

Estamos falhando por falta de metodologias mais adequadas e epistemologicamente mais atualizadas, inspiradas em paradigmas que facilitem a operacionalização dos trabalhos na direção construtiva e criativa que almejamos. Estamos falhando porque não estamos formando, adequada e oportunamente, as novas gerações para enfrentarem os desafios atuais,

² Esse fenômeno é denominado por Lemos (2006) como “liberação do polo da emissão”, o primeiro princípio da cibercultura.

³ Índice de Desenvolvimento da Educação Básica.

já que estamos educando com metodologias cientificamente defasadas, usando tecnologias que camuflam velhas teorias a partir de propostas que continuam vendo o aluno como um mero espectador, um simples receptor de estímulos, um eterno copador e reproduzidor de informações (MORAES, 2002, p. 3).

Esse excerto revela uma dura crítica à concepção empirista de educação e suas práticas pedagógicas instrucionistas, isto é, práticas que percebem o aluno como mero receptor de informações, e o professor como transmissor do conhecimento verdadeiro. De forma coerente com o conceito de educação construído até o momento, e em consonância com a crítica de Moraes, sugere-se que somente em um cenário em que os alunos participem ativamente da construção de seus conhecimentos, em vez de atuarem como meros receptáculos de informação, as tecnologias digitais poderão ser utilizadas no limite de suas potencialidades.

Finalmente, entendemos que não basta simplesmente inundar as escolas com TDICs sofisticadas como *serious games* ou quaisquer outros *softwares* educativos, ou com diferentes formas de pensar, como o pensamento crítico, computacional ou *design thinking*⁴. É necessário que a concepção de educação como um todo se transforme, no sentido de perceber o aluno como protagonista de seu processo de aprendizagem.

Por conseguinte, as lentes teóricas da abordagem *vigotskiana* de desenvolvimento humano parecem adequadas para observar os processos de desenvolvimento e aprendizagem no âmbito desta pesquisa, na qual se pretende identificar as contribuições dos *serious games* no desenvolvimento do pensamento computacional. Além disso, a teoria de Vigotski, que será descrita na próxima seção, foi escolhida principalmente devido às experiências prévias do autor.

2.2 Aprendizagem à luz da teoria vigotskiana

Lev Vigotski⁵, distinto filólogo, psicólogo e defectologista russo, é considerado um dos pioneiros na área da psicologia do desenvolvimento humano. Apesar de sua morte prematura, aos 38 anos, Vigotski produziu uma miríade de publicações nessas áreas, tendo investigado temas como a psicologia infantil, a psicologia da arte, a epistemologia da psicologia, a relação entre o pensamento e a linguagem e a educação de crianças com necessidades especiais.

A teoria de Vigotski é conhecida como histórico-cultural, pois objetiva “caracterizar os aspectos tipicamente humanos do comportamento e elaborar hipóteses de como essas

⁴ Forma de pensamento que engloba métodos e processos para a abordagem de problemas, sob diferentes ângulos e perspectivas, com enfoque no trabalho colaborativo em equipes multidisciplinares. Faz alusão ao pensamento dos *designers*.

⁵ Considerando a ausência de consenso quanto à grafia do nome do autor, optou-se por utilizar “Vigotski”. A grafia escolhida será utilizada em todos os casos, com exceção de citações diretas e referências.

características se formaram ao longo da história humana e de como se desenvolvem durante a vida de um indivíduo” (VIGOTSKI, 2007, p. 3). Um dos principais fundamentos dessa abordagem é a ideia de que a “internalização das atividades socialmente enraizadas e historicamente desenvolvidas constitui o aspecto característico da psicologia humana; é a base do salto quantitativo da psicologia animal para a psicologia humana” (p. 58).

A partir desse fundamento, percebemos que Vigotski atribui grande destaque às relações sociais e ao aspecto histórico da atividade humana em seus estudos. Por conta disso, sua perspectiva de desenvolvimento humano também é conhecida como sócio-histórica, sociocultural ou sociointeracionista. No contexto desta pesquisa, essa abordagem serviu como norteador teórico e como base para a identificação e análise das contribuições dos *serious games* no desenvolvimento do pensamento computacional.

Ao longo desta seção, serão apresentados alguns dos conceitos-chave presentes nas contribuições teóricas de Vigotski, como processos psicológicos superiores, mediação, internalização e zona de desenvolvimento proximal (ZDP). E, além de prover uma melhor compreensão dos termos utilizados pelo autor, pretendemos articular esses conceitos de tal modo que seja possível definir o que seja aprendizagem à luz de sua teoria — um empreendimento essencial para se responder à questão de pesquisa desta dissertação.

Antes disso, todavia, é imprescindível ressaltarmos que a obra de Vigotski tem sido revisada nas últimas décadas, tanto no exterior, quanto no Brasil. Tal esforço busca esclarecer alguns aspectos teóricos relacionados às traduções das primeiras obras que chegaram ao ocidente, em inglês, e ao chamado “culto à Vigotski”, como apontam os estudos de Prestes (2012), Cole (2009) e Yasnitsky e Veer (2015). O presente quadro teórico leva em consideração alguns dos argumentos dos revisionistas e, por conta disso, pode causar certa estranheza face aos conceitos vigentes na comunidade acadêmica.

2.2.1 Processos psicológicos superiores

Alguns dos termos mais recorrentes nos textos de Vigotski são os processos psicológicos superiores, também conhecidos como funções psicológicas superiores. Para Vigotski (2007), existem duas linhas qualitativamente diferentes de desenvolvimento psicológico humano: os processos elementares e os processos psicológicos superiores.

Os processos elementares são de origem biológica e estão relacionados principalmente às reações sensório-motoras (ver Quadro 1); não são mentais e nem sociais (RATNER, 1991). Conforme Vigotski (2007), as funções elementares se caracterizam principalmente por serem diretamente determinadas pelos estímulos do ambiente. Tais funções também estão presentes em algumas espécies de animais. Por exemplo, um chimpanzé, ao ver que uma vara foi colocada próximo à grade de sua jaula, examina-a e percebe que pode utilizá-la para alcançar um cesto de frutos que está do lado de fora. Nesse exemplo, o

que desencadeou a ação do chimpanzé foi um estímulo externo, proveniente do ambiente – a ideia de encontrar ou fabricar um instrumento para pegar a cesta de frutos não foi planejada pelo chimpanzé, mas surgiu de um imediatismo⁶, uma reação direta à sua experiência percebida de mundo.

Os processos psicológicos superiores, funções cognitivas particularmente humanas, têm origem sociocultural, isto é, nascem na relação entre o indivíduo e seu meio cultural e social (ver Quadro 1). Diferentemente das funções elementares, caracterizadas unicamente por reações imediatas a estímulos externos, as funções psicológicas superiores se caracterizam por serem mediadas, isto é, por pressupor um elo intermediário entre o estímulo e a resposta, o qual pode ser um instrumento ou um signo⁷ (VIGOTSKI, 2007). A diferença primordial entre esses é que o instrumento é orientado externamente, conduz à influência humana sobre o objeto de atividade e serve como meio para o controle da natureza, ao passo que o signo é orientado internamente, não modifica o ambiente e atua como meio de controle do próprio indivíduo (VIGOTSKI, 2007).

Quadro 1 – Processos psicológicos elementares e superiores

Elementares	Superiores
Sensação	Imaginação
Atenção	Consciência
Integração modal	Cognição
Percepção	Memória mediada/indireta
Memória	Atenção voluntária
Audição	Atividade mediada
Visão	Linguagem social (racional)
Somestesia	Pensamento
Olfato	Leitura
Emoção	Escrita
Linguagem animal (emotiva)	
Interocepção	
Propriocepção	
Exterocepção	

Fonte: adaptado de (MORAES, 2010).

⁶ No âmbito das obras de Vigotski, o termo imediato quase sempre significa aquilo que é direto, não mediado, sem interferências, normalmente contrastando com a noção de mediação, o acesso a recortes da realidade por intermédio de instrumentos ou signos.

⁷ Para Vigotski (1930), signos (instrumentos psicológicos) são “formações artificiais... [que] são sociais, não orgânicas ou individuais”. Alguns exemplos são: “a linguagem; vários sistemas de contagem; técnicas mnemônicas; sistemas de símbolos algébricos; obras de arte; a escrita; esquemas, diagramas, mapas e desenhos mecânicos; todos os tipos de sinais convencionais”. De forma mais geral, um signo pode ser qualquer objeto, forma ou fenômeno que representa algo diferente de si mesmo.

2.2.2 Mediação

A concepção de que o homem acessa e transforma a realidade por meio de elos intermediários, os instrumentos e signos, é conhecida como mediação – a base dos processos psicológicos superiores. A partir dessa perspectiva, sobre a qual se alicerça a teoria *vigotskiana*, “enquanto sujeito do conhecimento, o homem não tem acesso direto aos objetos, mas um acesso mediado, isto é, feito através de recortes do real, operados pelos sistemas simbólicos de que dispõe” (OLIVEIRA, 2016, p. 26). Em outras palavras, o ser humano, dotado de funções psicológicas superiores como cognição, linguagem, memória mediada, imaginação e atenção voluntária, não simplesmente age sobre os estímulos do ambiente; antes disso, “ativamente modifica esses estímulos e usa suas modificações como um instrumento de seu comportamento” (COLE; LEVITIN; LURIA, 2005, p. 44, tradução do autor)⁸. Ou seja, o ser humano planeja sua ação no mundo com base nos instrumentos e sistemas simbólicos de que dispõe, a partir de construções e aquisições culturais.

Na contracapa do livro “A formação social da mente: o desenvolvimento dos processos psicológicos superiores”, que reúne textos selecionados de Vigotski, há uma passagem onde se pode encontrar uma noção de mediação, ainda que de forma embrionária:

A aranha realiza operações que lembram o tecelão, e as caixas suspensas que as abelhas constroem envergonham o trabalho de muitos arquitetos. Mas até mesmo o pior dos arquitetos difere, de início, da mais hábil das abelhas, pelo fato de que antes de fazer uma caixa de madeira, ele já a construiu mentalmente. No final do processo do trabalho, ele obtém um resultado que já existia em sua mente antes de ele começar a construção. O arquiteto não só modifica a forma que lhe foi dada pela natureza, dentro das restrições impostas por ela, como também realiza um plano que lhe é próprio, definindo os meios e o caráter da atividade aos quais ele deve subordinar sua vontade. (MARX apud VIGOTSKI, 2007, contracapa)

Nesse excerto, Karl Marx – que influenciou Vigotski significativamente – exemplifica como a inteligência humana se distingue da animal, destacando o fato de que, antes de agir, o ser humano opera e planeja mentalmente, utilizando meios construídos por ele mesmo. De acordo com Cole, Levitin e Luria (2005), esse foi um dos conceitos-chave na construção do quadro teórico de Vigotski, pois, influenciado por essa ideia marxiana, ampliou o conceito de mediação e, conseqüentemente, desenvolveu a base para uma psicologia experimental das funções psicológicas superiores.

Ainda na perspectiva de Vigotski, mas numa interpretação contemporânea de sua psicologia sociocultural, Ratner (1991) concebe três tipos distintos de mediação: a) a consciência ou atividade mental, b) a cooperação social que inclui a linguagem e as interações e c) os instrumentos, que incluem a tecnologia. Para o autor, os instrumentos são acessórios utilizados para ampliar as capacidades naturais do organismo físico, sendo,

⁸ “[...] actively modifies those stimuli and uses his modifications as an instrument of his behavior.”

portanto, extensões do homem no mundo. Os *serious games*, um dos temas desta pesquisa, incluem-se neste último tipo de mediação.

2.2.3 Internalização

Conforme Vigotski (2007) afirma, seria um enorme erro acreditar que as operações mediadas, características essenciais das funções psicológicas superiores, aparecem como decorrência de pura lógica, como um lampejo repentino. A criança não deduz intuitivamente ou descobre as relações entre os signos e as formas de utilizá-los, como se desenvolvesse uma “atitude abstrata derivada, por assim dizer, das ‘profundezas de sua própria mente’” (VYGOTSKY, 1978, p. 45, tradução do autor)⁹.

Negando uma possível visão metafísica que conceba um esquema psicológico pré-existente, a qual implicaria na existência a priori dos processos psicológicos superiores, Vygotsky (1978, p. 45) defende que as operações com signos aparecem como resultado de longos e complexos processos de evolução psicológica. Ressaltamos, todavia, que Vigotski jamais renunciou as bases biológicas das funções psicológicas superiores; em vez disso, ele argumenta que a base biológica é mais um substrato geral que oferece possibilidades do que um determinante específico do funcionamento cerebral (RATNER, 1991).

De acordo com Vigotski (2007), as funções psicológicas superiores ocorrem inicialmente no campo interpsicológico, e se deslocam para o campo intrapsicológico na medida em que o indivíduo assimila os instrumentos e sistemas simbólicos provenientes da cultura que o permeia. Em outras palavras, os processos psicológicos superiores primeiro se desenvolvem externamente e, num segundo momento, internamente. Essa transição do domínio interpsicológico para o intrapsicológico é chamada de *internalização* ou *interiorização*.

A internalização consiste na reconstrução interna (individual) de operações externas (socioculturais e históricas) (VIGOTSKI, 2007, p. 56). Essas operações externas são as formas culturais de comportamento, as quais podem ser representadas mentalmente (internamente), por meio de signos. Assim, tão logo ocorre a internalização, torna-se possível operar mentalmente sem a presença dos referentes concretos, uma vez que foi criada uma representação simbólica para eles (OLIVEIRA, 2016).

2.2.4 Aprendizagem

Um dos conceitos mais importantes no âmbito desta pesquisa é aprendizagem¹⁰. Conforme ressalta Prestes (2012), a noção de aprendizagem encontrada nos textos tra-

⁹ “[...] abstract attitude derived, so to speak, from ‘the depths of the child’s own mind.’”

¹⁰ Ao longo deste texto, utiliza-se o verbete *aprendizagem* para se referir ao ato de aprender em detrimento da palavra *aprendizado*. Essa escolha foi tomada por se acreditar que *aprendizado* remete àquilo que já foi aprendido, como um produto, enquanto *aprendizagem* denota um processo vitalício, o que é mais coerente com a abordagem *vigotskiana* que permeia esta dissertação.

duzidos de Vigotski é uma aproximação da palavra *obučenie*, a qual corresponde aos seguintes verbetes da língua portuguesa: ensino, instrução, treinamento, educação, curso e aprendizagem.

Michael Cole, um dos principais especialistas da obra de Vigotski, aponta que há mais do que certa ambiguidade no emprego da palavra aprendizagem nos textos traduzidos do autor, principalmente no capítulo “Interação entre aprendizado e desenvolvimento”, do livro “A formação social da mente” (*Mind in society*, em inglês), organizado por ele mesmo. Para o leitor desavisado, esse texto pode levar a entendimentos errôneos, uma vez que o significado da palavra aprendizagem não é equivalente à sua acepção na língua inglesa contemporânea. Cole assevera que *obučenie* se refere a um

processo bilateral, no qual um lado de fato se refere à aprendizagem (uma mudança nos processos psicológicos e no conhecimento da criança), mas o outro se refere à organização do ambiente pelo adulto, o qual, assume-se no artigo em discussão, é um professor em uma escola formal com o poder sobre a organização da experiência da criança. Então, naquele contexto, com aqueles propósitos, *obučenie* é mais adequadamente traduzido como “instrução” ou “ensino”. (COLE, 2009, p. 292, tradução do autor)¹¹

Observando tais nuances apontados por Cole na obra de Vigotski, podemos entender que a aprendizagem é um processo bilateral que se refere tanto às mudanças nos processos psicológicos e no conhecimento do aprendiz, quanto ao esforço empreendido pelo educador na organização da experiência de aprendizagem (ensino). Assim, é possível associar o ato de aprender ao de internalizar, pois ambos pressupõem interação entre o sujeito aprendiz com um outro mais experiente (*more knowledgeable other* – MKO).

2.2.4.1 Relações entre aprendizagem e desenvolvimento

Ao estudar a interação entre aprendizagem e desenvolvimento, Vigotski partiu do pressuposto de que esses processos não são coincidentes, mas que eles estão diretamente inter-relacionados desde o primeiro dia de vida (VIGOTSKI, 2007, p. 95). Vigotski (2007, p. 103) postula também que a aprendizagem impulsiona processos internos de desenvolvimento, os processos psicológicos superiores.

Se, como mencionado anteriormente, a internalização de atividades desenvolvidas social e historicamente constitui o aspecto característico da psicologia humana, e esse aspecto remete às funções psicológicas superiores, deduzimos que tais funções se constituam pela internalização. Por conseguinte, se tanto o processo de aprendizagem quanto o de

¹¹ “double-sided process, one side of which does indeed refer to learning (a change in the psychological processes and knowledge of the child), but the other of which refers to the organization of the environment by the adult, who, it is assumed in the article under discussion, is a teacher in a formal school with power over the organization of the children’s experience. So in that context, for those purposes, *obuchenie* is most adequately translated as ‘instruction’ or ‘teaching’”

internalização resultam no desenvolvimento das funções psicológicas superiores, podemos dizer que esses processos são quase equivalentes, pois expressam sentidos muito próximos.

O aprendizado desperta vários processos internos de desenvolvimento, que são capazes de operar somente quando a criança interage com pessoas em seu ambiente e quando em cooperação com seus companheiros. Uma vez internalizados, esses processos tornam-se parte das aquisições do desenvolvimento independente da criança. (VIGOTSKI, 2007, p. 103)

Nesse sentido, se aprender é internalizar e a internalização consiste em reconstruir internamente as formas histórico-culturais de comportamento, então, a aprendizagem depende das interações dos indivíduos com o meio sociocultural. Essa dedução é coerente com a concepção de mediação, como corrobora Vigotski, ao afirmar que o “caminho do objeto até a criança e desta até o objeto passa por outra pessoa” (VIGOTSKI, 2007, p. 20).

Isso implica em dizer que o professor exerce uma função vital no processo de aprendizagem, uma vez que, na condição de “outra pessoa mais experiente” (*more knowledgeable other* – MKO), ele está guarnecido de um maior arcabouço cultural e deve organizar a experiência de seus alunos. Na concepção bilateral de aprendizagem de Vigotski, como aponta Cole (2009), o professor atua no lado do ensino, da instrução. A partir de suas intervenções pedagógicas como mediador, os alunos têm acesso ao material simbólico por ele fornecido e têm a possibilidade de dar início a novos processos de internalização.

Assim, entendemos que a aprendizagem *vigotskiana* consiste num processo ativo e consciente¹² de mudança nos processos psicológicos superiores e de aquisição de formas histórico-culturais de comportamento (conhecimento culturalmente desenvolvido e enraizado), o qual pressupõe um elo intermediário entre o sujeito e o objeto, ou seja, é uma *aprendizagem mediada*.

Recentemente, o conceito de aprendizagem mediada tem sido utilizado em pesquisas relacionadas à educação, softwares educativos e jogos digitais (SANTOS, 2012; SALVADORI, 2016; POLONI, 2018; SEVERGNINI; SOARES, 2019). Nesse contexto, a mediação se refere principalmente à atuação do professor como alguém que mostra o caminho até o objeto de conhecimento para o sujeito. Também é comum a utilização do termo “elemento mediador” para se referir aos artefatos culturais que garantem o acesso ao objeto de conhecimento, como livros, revistas, áudios, vídeos, conteúdos multimídia, hipertexto, *software*, jogos etc.

¹² Com Ratner (1991), estudioso do trabalho de Vigotski, entendemos que a atividade é ativa e consciente, e que ela transcende as simples associações estímulo-resposta (S-R) das teorias *behavioristas* de aprendizagem, repudiadas por Vigotski.

2.2.5 Zona de desenvolvimento proximal

Em seus estudos acerca da interação entre aprendizagem e desenvolvimento, Vigotski (2007) observou que crianças com o mesmo nível de desenvolvimento mental¹³ apresentavam capacidades de aprender muito distintas sob a orientação de alguém mais experiente. Dessa forma, Vigotski percebeu que, na realidade, elas não tinham a mesma idade mental e que o curso subsequente de seus aprendizados seria diferente. A partir disso, ele concluiu que era necessária uma nova abordagem para estimar o nível de desenvolvimento mental, que permitisse avaliar o processo de desenvolvimento, não somente seus produtos.

Para esse fim, Vigotski (2007) criou o conceito de Zona de Desenvolvimento Proximal (ZDP), que pode ser definido como

a distância entre o *nível de desenvolvimento real*, que se costuma determinar através da solução independente de problemas, e o *nível de desenvolvimento potencial*, determinado através da solução de problemas sob a orientação de um adulto ou em colaboração com companheiros mais capazes (VIGOTSKI, 2007, p. 97 – grifos do autor)

Atualmente, a noção de ZDP tem sido utilizada em diversos projetos de pesquisa relacionados à educação, TDICs e jogos, abrangendo temáticas como: gamificação (FARDO, 2013), pensamento computacional e matemático (BARCELOS, 2014), RPG (*role-playing game* ou jogo de representação de papéis) (JAQUES, 2016), *software* de teoria e percepção musical (SALVADORI, 2016) e *serious games* (SEVERGNINI, 2018; SEVERGNINI; SOARES, 2019). Com base nesse conceito, como se observa nessas pesquisas, educadores, *game designers* e *designers* instrucionais têm potencializado as oportunidades de aprendizagem dos alunos.

Vigotski (2007) defende que a ZDP permite estimar com maior precisão o nível de desenvolvimento de uma criança, pois considera tanto as funções desenvolvidas quanto as que ainda estão em processo de maturação. Para Vigotski, isso é muito importante, pois os dois níveis de desenvolvimento são dinâmicos: o que hoje faz parte do *desenvolvimento potencial*, sendo realizado apenas com a participação de alguém mais experiente, amanhã será *desenvolvimento real*, tornando-se parte das aquisições do indivíduo, por meio da internalização. Assim, entende-se que o nível de desenvolvimento mental de uma criança somente pode ser estimado se levados em consideração seus dois níveis: o desenvolvimento real e a ZDP (VIGOTSKI, 2007).

Nessa perspectiva, a aprendizagem, no sentido *vigotskiano* da palavra, tem extrema importância no desenvolvimento dos processos psicológicos superiores. “O aprendizado adequadamente organizado resulta em desenvolvimento mental e põe em movimento

¹³ No contexto do estudo, os níveis equivalentes foram reconhecidos via testes de Q.I. padronizados.

vários processos de desenvolvimento que, de outra forma, seriam impossíveis de acontecer” (VIGOTSKI, 2007, p. 103)¹⁴.

No entanto, não são todas as práticas pedagógicas ou problemas que conseguem perturbar a ZDP, isto é, impulsionar os processos de desenvolvimento. Conforme Vigotski (2007), as atividades propostas devem se adiantar ao desenvolvimento real e interferir no nível de desenvolvimento potencial do indivíduo¹⁵. Por conseguinte, os problemas propostos aos alunos não podem nem ser fáceis a ponto de não perturbar suas ZDP (ex: ensinar algum conteúdo que os alunos dominam), nem difíceis a ponto de extrapolar completamente seus níveis de desenvolvimento potencial (ex: demonstrar a transformada de Fourier para alunos que estão aprendendo aritmética). Assim, percebe-se que a ZDP constitui um importante meio para auxiliar o professor na criação de estratégias e intervenções mediadoras.

Finalmente, ressaltamos que o arcabouço teórico construído sobre as noções de ZDP, processos psicológicos superiores, mediação, sociointeração, MKO (outro mais experiente) e internalização, conceitos vigotskianos fortemente relacionados ao processo de aprendizagem, foi de suma importância para a pesquisa como um todo.

2.3 *Serious games*

Serious games – jogos sérios, em português – são jogos que não tem entretenimento, prazer ou diversão como propósito primário (MICHAEL; CHEN, 2005). Ritterfeld, Cody e Vorderer (2009) definem que *serious games* são jogos desenvolvidos com conteúdo e propósito além do puro entretenimento, caracterizando-se normalmente como jogos para ensinar, treinar e promover hábitos saudáveis e mudança social. No contexto desta pesquisa, tenciona-se investigar as contribuições dos *serious games* como elementos mediadores do desenvolvimento do pensamento computacional.

O termo “*serious games*”, em sua acepção atual, apareceu pela primeira vez no livro homônimo escrito por Clark C. Abt, em 1970. Nesse livro, Abt dá uma definição clara para esse tipo de jogo:

Jogos podem ser jogados a sério ou casualmente. Estamos preocupados com os jogos sérios no sentido de que esses jogos têm um propósito educacional explícito e cuidadosamente pensado e não se destinam a serem jogados fundamentalmente pela diversão. Isso não significa que os jogos sérios não sejam, ou não devam ser, divertidos. (ABT, 1987, p. 9, tradução do autor)¹⁶

¹⁴ Esse é um exemplo clamoroso para ilustrar os equívocos de tradução da obra de Vigotski. Neste caso, em específico, entende-se que o sentido se aproxima mais de ensino do que de aprendizagem.

¹⁵ Essa é uma noção um tanto quanto óbvia, mas que precisa ser pontuada: não faz sentido ensinar a uma pessoa aquilo que ela já conhece, então é necessário dialogar com ela para entender de que ponto deve-se partir.

¹⁶ “Games may be played seriously or casually. We are concerned with *serious games* in the sense that these games have an explicit and carefully thought-out educational purpose and are not intended to be

Como destaca Abt, o evidente oxímoro – jogo sério – não implica que esses jogos não sejam interessantes, prazerosos ou divertidos; a diferença em relação aos outros jogos é que eles possuem outro propósito, uma motivação ulterior. Ben Sawyer (apud MICHAEL; CHEN, 2005), cofundador da *Serious Game Initiative*, por sua vez, explica que a palavra sério em *serious games* tem a intenção de refletir o propósito do jogo, o porquê de sua criação, não necessariamente se refere ao conteúdo do jogo *per se*.

Em consonância com as percepções de Gee (2003; 2005) e Prensky (2003), Ritterfeld e Weber (2006) acrescentam que qualquer jogo digital pode apresentar formas de aprendizagem, independentemente de o jogo ser considerado sério ou não. Nesse aspecto, o que diferencia um jogo digital comercial de um jogo sério é que os *serious games* são desenvolvidos com a colaboração de *game designers*, educadores, pedagogos, *designers* instrucionais e especialistas em outras áreas do conhecimento, e com uma finalidade que transcende o puro entretenimento. Além disso, os *serious games* não costumam apresentar elementos impróprios ou negativos, como a representação de violência contra indefesos, o genocídio e a glamorização do consumo de drogas, por exemplo, como acontece de forma recorrente nos jogos comerciais.

2.3.1 Possibilidades e limitações

O educador e pesquisador francês Damien Djaouti (2011; 2016), cuja tese de doutoramento o tornou referência na área dos jogos sérios, defende que as possibilidades e limitações dos *serious games* dependem de cada jogo e do contexto nos quais eles são utilizados. Em outras palavras, não é possível generalizar; um jogo sério pode ter qualidade próxima ou superior a dos jogos comerciais bem-sucedidos, ter excelentes dinâmicas e mecânicas de jogo e alta qualidade audiovisual, ao mesmo tempo em que pode ter uma qualidade visivelmente inferior, com mecânicas pouco ou nada atrativas, ser esteticamente disforme e até mesmo possuir *bugs*¹⁷. O contexto também é fator determinante do potencial educativo de um *serious game*: se o propósito do jogo se alinha à proposta de uma disciplina, ou se ele cria a possibilidade de novas disciplinas, é provável que o jogo seja significativo; caso contrário, se o jogo for utilizado apenas como atrativo tecnológico ou for incompatível com a proposta pedagógica, dificilmente serão realizadas as significações necessárias para a aprendizagem.

Embora não seja possível generalizar, os *serious games* compartilham uma série de possibilidades e limitações. As principais possibilidades dos *serious games* são o aumento da motivação dos alunos, a aprendizagem por tentativa e erro, a consideração das diferenças

played primarily for amusement. This does not mean that *serious games* are not, or should not be, entertaining.”

¹⁷ Termo utilizado para se referir aos defeitos, falhas, erros e comportamentos inesperados em um software. A palavra *bug* passou a ter essa conotação devido aos problemas de funcionamento ocasionados por insetos no ENIAC, um dos primeiros computadores completamente eletrônicos.

de ritmo de aprendizagem e o incentivo das interações pedagógicas entre os alunos (PIVEC; PIVEC, 2009).

Coerente com as ideias de Vigotski (2007), destaca-se que a motivação se apresenta como um fator muito importante nos processos de ensino e aprendizagem. Whitton (2010) aponta que um dos principais fatores para esse aumento significativo na motivação dos jogadores é o *feedback* regular que os jogos fornecem para cada ação desempenhada, isto é, os controles enviados pelo jogador afetam quase que instantaneamente o estado do jogo.

Assim, dependendo das diretrizes de *design* de um jogo, a noção de *feedback* pode ser tanto associada ao behaviorismo quanto ao construtivismo. No clássico jogo de quebra-cabeças Tetris¹⁸, por exemplo, a principal mecânica (regra) de jogo exige que o jogador gire as formas geométricas, que caem constantemente do topo da tela, e as posicione da melhor forma possível em sua base, com o intuito de formar linhas que atravessem horizontalmente todo o cenário, de modo a eliminar os blocos geométricos envolvidos e gerar mais espaço para dar continuidade ao processo. Desde o primeiro segundo de cada partida, o jogador precisa atualizar constantemente sua estratégia com base nas formas geométricas geradas aleatoriamente pelo jogo. Assim, é possível interpretar que o jogador desempenha principalmente um papel reativo, pois apenas reage àquilo que se lhe apresenta (é claro que ele pode antecipar às próximas peças a serem geradas, mas isso exige um nível muito maior de habilidade do jogador).

*No Man's Sky*¹⁹, por sua vez, um jogo de ação-aventura, o qual envolve exploração e sobrevivência em um universo com mais de 18 quintilhões de planetas, exige um papel muito mais ativo dos jogadores. Em *No Man's Sky*, se o jogador não fizer absolutamente nada, o jogo tenderá a não apresentar nenhum *feedback*. Por outro lado, conforme esse jogador explora os planetas e constrói sua própria experiência, situações que exigem novas intervenções lhe são apresentadas. Portanto, a ideia de *feedback per se* não denota a forma em que um jogo percebe os jogadores (passivos ou ativos); isso varia não somente de um jogo para outro, mas também em diferentes situações de um mesmo jogo. Por isso, a forma na qual se fornece *feedback* deve ser levada em consideração no momento da escolha do *serious game*, se há a intenção de se manter a coerência teórico-epistemológica. De qualquer forma, é o *feedback* dos jogos eletrônicos que motiva os jogadores a interagir.

A segunda principal possibilidade dos *serious games* é a aprendizagem por tentativa e erro (PIVEC; PIVEC, 2009), que está fortemente vinculada às noções de interação e *feedback*. Os jogos sérios oferecem um espaço de experimentação para os alunos, onde eles são convidados a pensar e construir mentalmente hipóteses para solucionar os problemas enfrentados. Sanchez (2011) corrobora ao afirmar que um bom jogo sério deve oferecer informações ao jogador para ajudá-lo a construir uma hipótese correta. Em outras palavras,

¹⁸ Fonte: <<https://en.wikipedia.org/wiki/Tetris>>

¹⁹ Fonte: <https://en.wikipedia.org/wiki/No_Man's_Sky>

um *serious game* deve fornecer informações, instruções e *feedback* que permitam ao jogador reformular suas hipóteses²⁰ até que consiga vencer. Essa abordagem por tentativa e erro se dá, principalmente, porque os jogos eletrônicos permitem que a aprendizagem ocorra sem as consequências do mundo real (MICHAEL; CHEN, 2005), o que pode ser visto como benéfico, uma vez que isso tende a diminuir o medo de errar por parte dos jogadores (FARDO, 2013; SEVERGNINI, 2018; SEVERGNINI; SOARES, 2019).

A terceira possibilidade dos *serious games* reside na consideração das diferenças de ritmo de aprendizagem (PIVEC; PIVEC, 2009). Isso significa que cada aluno pode progredir no jogo em seu próprio ritmo. Alguns alunos podem repetir dezenas de vezes o mesmo desafio, até que consigam entendê-lo, sem se preocupar com o julgamento de seus colegas; ao mesmo tempo, os alunos que obtiverem sucesso em poucas tentativas não precisam aguardar pelos demais.

A quarta e última das principais possibilidades dos *serious games* é o incentivo às interações pedagógicas entre os alunos. Jogos com modalidades multijogador podem estimular interações educacionais entre jogadores (DJAOUTI, 2016); isso possibilita a troca de experiências entre pares com níveis de desenvolvimento intelectual diferentes e, conseqüentemente, propicia a mobilização da ZDP (VIGOTSKI, 2007; DJAOUTI, 2016), uma vez que os alunos mais experientes podem ajudar seus colegas.

No que se refere às limitações dos *serious games*, Pivec e Pivec (2009) citam o uso de jogos sérios irrelevantes, a falta de integração do jogo no trabalho do professor e as restrições materiais e logísticas. Wastiau et al. (2009) apontam que a limitação mais importante é a adequação da escolha do jogo a ser usado. Conforme mencionado anteriormente, nem todos os jogos sérios são pertinentes e isso se deve principalmente a fatores como contexto e qualidade. Além disso, no que diz respeito à concepção de aprendizagem, existem diferentes tipos de *serious game* (behaviorista, construtivista etc.) – como visto na comparação entre *Tetris* e *No Man's Sky*. Destacamos, então, a importância da seleção cuidadosa do *serious game*, pelo professor, pensando na coerência em relação aos seus alunos, objetivos educacionais e seus métodos de trabalho (DJAOUTI, 2016). Afinal, são as estratégias de ensino que ditam as tecnologias utilizadas, e não o contrário.

O papel do professor começa na escolha do *serious game*, mas não deveria se limitar à ela. Essa é a segunda principal limitação dos jogos sérios: a falta de integração do jogo no trabalho do professor. De acordo com vários estudos sobre o uso efetivo dos jogos sérios em sala de aula (HABGOOD, 2007; HOCHET, 2011; DJAOUTI, 2016), o professor não pode ficar alheio ao jogo, pois ele desempenha uma função essencial no sucesso dessa atividade. Para Hochet (2011), professor de história e geografia no ensino superior, a mera utilização de um jogo não parece interferir muito no que os alunos aprendem; por conta disso, a

²⁰ Colocado desta forma, pode parecer que esse processo ocorre de forma mecânica; porém, na experiência de jogo esse processo ocorre quase que imperceptível, tamanha a fluidez da interação.

mediação do professor, que pensa o lugar do jogo no processo de aprendizagem, continua sendo essencial. Habgood (2007) ressalta que um mesmo *serious game* pode ser muito mais efetivo para a aprendizagem se houver a intervenção do professor. Esse argumento será retomado mais adiante, no contexto da teoria *vigotskiana*.

Por fim, destacam-se as restrições materiais e logísticas, as quais se referem à disponibilidade de computadores ou consoles (dispositivos de jogo eletrônico) com capacidade de executar os jogos, bem como ao acesso à internet. Atualmente, devido ao maior acesso a computadores nas escolas, essa limitação tem se tornado cada vez menos preocupante. Apesar disso, uma parte dos jogos online e/ou pagos ainda está inacessível para determinadas instituições escolares.

Dito isso, percebemos que existe uma série de possibilidades que podem ser aproveitadas por meio da utilização dos *serious games*. Ao mesmo tempo, esse gênero de jogo possui certas limitações. Conhecer tais limitações pode ser tão importante quanto saber de suas potencialidades, pois, apesar de eficazes transmissores de conhecimento, jogos sérios não são ferramentas mágicas; tampouco substituem o professor (DJAOUTI, 2016). É importante que o docente se aproprie desse potente suporte pedagógico antes de utilizá-lo.

2.4 Pensamento computacional

Pensamento Computacional (PC) é um termo cuja criação costuma ser atribuída à cientista da computação Jeannette Wing, devido a uma publicação de sua autoria, no ano de 2006. Nessa publicação, Wing (2006) descreveu o PC como uma forma de raciocínio que utiliza conceitos e técnicas da ciência da computação para a resolução de problemas em diferentes contextos, não se limitando aos especialistas da área. Desde então, o termo ganhou notoriedade na comunidade acadêmica.

No entanto, a expressão “pensamento computacional” foi mencionada pelo menos uma vez antes dessa ocasião, por Seymour Papert, um dos pioneiros na investigação da aprendizagem mediada por computadores, em seu livro “A máquina das crianças”. Em um dos capítulos desse livro, Papert (2008) se referiu ao PC, ao explicar a importância da precisão matemática nos cálculos dos mísseis da Segunda Guerra Mundial. Devido à grande quantidade de variáveis envolvidas no cálculo da trajetória dos projéteis (condições atmosféricas, posições dos alvos etc.), matemáticos célebres como John von Neumann e Norbert Wiener foram mobilizados para auxiliar na tarefa. Aos poucos, os esforços mentais dos matemáticos se mostraram insuficientes, tamanha era a demanda por tais estimativas. Para resolver esse problema, foram projetados os primeiros computadores, máquinas capazes de realizar milhares de operações matemáticas por segundo. Desde então, os cálculos passaram a ser elaborados de uma maneira que pudesse ser computada por tais máquinas.

Mesmo que implicitamente, a noção de PC apresentada por Papert (2008), ainda no contexto da Segunda Guerra, não se distancia daquela apresentada inicialmente por Wing, pois se refere a uma forma de raciocínio que já considera os limites computacionais do hardware para a resolução de problemas. Logo, podemos sugerir que o conceito de PC não é novo: ele antecede temporalmente o termo que o designa, uma vez que se evidencia desde os tempos de Von Neumann, George Boole e Alan Turing, alguns dos principais nomes da história da Computação – e, antes destes, em Babbage, Ada Lovelace, Pascal... Ao mesmo tempo em que seria injusto não creditar a Wing a criação do termo PC, considerando seus esforços para dar início às discussões acerca desse tema, seria ainda mais injusto não fazer alusão aos pioneiros da computação.

Isso posto, cabe ressaltarmos que a publicação original de Wing (2006, p. 35) contribuiu para caracterizar o que é PC e o que ele não é. A partir disso, algumas das principais características elencadas pela autora incluem:

- Conceituação, não programação: pensamento computacional não é apenas programação de computadores, pois pensar como um cientista da computação requer mais do que saber programar; é necessário pensar em múltiplos níveis de abstração;
- Uma maneira na qual humanos, não computadores, pensam: pensamento computacional é uma das formas nas quais os humanos resolvem problemas. Não é pensar como máquinas, pois máquinas não sabem pensar; são os humanos que a tornam úteis e interessantes;
- Complementa e se combina com o pensamento matemático e com a engenharia: a ciência da computação lança mão do pensamento matemático, pois tem como base a matemática. Ao mesmo tempo, a ciência da computação utiliza-se de engenharia, uma vez que sua aplicação resulta em sistemas que interagem com recursos do mundo real. “As restrições inerentes dos dispositivos computacionais forçam os cientistas da computação a pensar computacionalmente, não apenas matematicamente” (p. 35, tradução do autor²¹). Ademais, a liberdade para criar mundos virtuais possibilita a engenharia de sistemas além do mundo físico.

A partir dessas características, percebemos que pensar computacionalmente não é pensar como um computador, mas em formas de utilizar o poder computacional dele para resolver problemas. Ao mesmo tempo, o PC não se resume à programação, apesar de essa ser uma de suas principais aplicações práticas, como corroboram as pesquisas que

²¹ “The constraint of the underlying computing device force computer scientists to think computationally, not just mathematically”.

fazem uso do ambiente de programação Scratch²² para o ensino do PC (RESNICK, 2012; BARCELOS, 2014; POLONI, 2018).

Mais recentemente, Wing (2014) forneceu uma definição mais precisa para o conceito de PC: “o pensamento computacional é o processo de pensamento envolvido na formulação de um problema e na expressão de sua(s) solução(ões) de tal forma que um computador – humano ou máquina²³ – possa efetivamente executar” (WING, 2014, tradução do autor²⁴). Em consonância com essa definição, Karl Beecher, cientista da computação e autor de um dos primeiros livros dedicados ao tema, defende que:

O pensamento computacional é uma abordagem para solução de problemas que envolve o uso de um conjunto de práticas e princípios da ciência da computação para formular uma solução que seja executável por um computador. Não é só para programadores. Na verdade, é aplicável em uma variedade diversificada de campos. (BEECHER, 2017, p. 11, tradução do autor²⁵).

Baseado nas publicações dos principais autores do tema, Beecher não somente fornece uma definição para o PC como elenca quais seriam os principais conceitos da ciência da computação associados a essa forma de pensamento. Para o autor, os conceitos fundamentais relacionados ao PC são: o pensamento lógico, o pensamento algorítmico, a decomposição, a generalização e reconhecimento de padrões, a modelagem, a abstração e a avaliação (ver Quadro 2) (BEECHER, 2017).

Quadro 2 – Conceitos relacionados ao pensamento computacional

Conceito	Definição resumida
Pensamento lógico	Aplicação dos princípios da lógica para determinar a veracidade de um argumento (uma linha de raciocínio para chegar a uma conclusão).

(continua)

²² Scratch é um projeto do MIT que consiste num ambiente de programação de computadores com vistas à aprendizagem criativa de crianças e jovens. Para mais informações, acessar: <https://scratch.mit.edu>.

²³ Nessa definição, a autora dá a entender que também considera o ser humano um computador – o que faz sentido, dada sua capacidade de calcular. Isso vai ao encontro da história descrita por Papert, ao se referir aos “computadores” humanos Von Neumann e Norbert Wiener.

²⁴ “Computational thinking is the thought processes involved in formulating a problem and expressing its solution(s) in such a way that a computer—human or machine—can effectively carry out.”

²⁵ “Computational thinking is an approach to problem -solving that involves using a set of practices and principles from computer science to formulate a solution that’s executable by a computer. It’s not just for programmers. In fact, it’s applicable in a diverse array of fields”.

(continuação)

Conceito	Definição resumida
Pensamento algorítmico	Utilização do pensamento lógico para formular algoritmos (sequências de etapas claramente definidas que descrevem um processo composto por um conjunto finito de instruções não-ambíguas).
Decomposição	Abordagem que procura dividir um problema complexo em partes mais simples e mais fáceis de lidar (dividir e conquistar).
Generalização e reconhecimento de padrões	Reconhecimento de padrões em diferentes problemas (e subproblemas), de modo a generalizar uma solução aplicável para problemas similares ou um conjunto de problemas (simplificação e reutilização).
Abstração	Capacidade de compreender ideias ou conceitos fora de seu contexto concreto/específico. É provavelmente o conceito mais importante para o PC (WING, 2014).
Modelagem	Elaboração de modelos, isto é, representações abstratas de objetos do mundo real; abstrações (ex: não é possível transportar magicamente um problema do mundo real para o computador; é necessário criar modelos que contenham apenas informações realmente indispensáveis para a resolução do problema; a partir desses modelos, pode-se, finalmente, descrever uma solução que o computador consiga processar).
Avaliação	Avaliação de aspectos como exatidão, eficiência, elegância e usabilidade das soluções de um problema.

Fonte: adaptado de Beecher (2017).

Em conjunto com as características apresentadas por Wing (2006), os conceitos elencados por Beecher (2017) serviram como norteadores teóricos do PC no âmbito desta

pesquisa. Apesar de existirem outras referências na área, como as associações ISTE²⁶ e CSTA²⁷, Beecher foi escolhido por ter construído um consistente, preciso e atualizado referencial teórico para o PC em seu livro *Computational thinking: A beginner's guide to problem-solving and programming*.

2.4.1 Pensamento computacional? Por que não matemática ou lógica formal?

Um questionamento bastante recorrente em relação ao PC é como ele se relaciona com o raciocínio lógico-matemático. Isso tem sentido, uma vez que, historicamente, o PC – e toda a lógica computacional – nascem do arcabouço teórico da matemática. Entretanto, o conjunto de conceitos compreendidos no PC não se limita ao pensamento lógico e algorítmico; o PC também abarca as noções de decomposição, generalização e reconhecimento de padrões, abstração, modelagem e avaliação (BEECHER, 2017).

Ainda assim, é possível argumentarmos que tais conceitos não são exclusivos ao PC, pois eles também podem ser encontrados no pensamento matemático e nas ações próprias da área das Engenharias. Por conta disso, para diferenciar PC – e o conjunto de conceitos que o constitui – de outras formas de pensamento, é necessário retornar para a sua definição: “o pensamento computacional é o processo de pensamento envolvido na formulação de um problema e na expressão de sua(s) solução(ões) de tal forma que um computador – humano ou máquina – possa efetivamente executar” (WING, 2014). O PC é uma forma de pensamento que implica o uso de computadores como uma extensão da capacidade humana de resolver problemas.

Estabelecidas as diferenças, ainda resta justificar o porquê da escolha do pensamento computacional em detrimento da matemática ou da lógica formal. De acordo com Valente (2016), o PC é um tema que tem recebido atenção nos currículos do ensino básico de diversos países ao redor do mundo, incluindo o Brasil. Além do aspecto curricular, entretanto, existe uma justificativa epistemológica para a investigação do PC. Conforme aponta Blikstein:

estamos ensinando nossos alunos que a tecnologia serve para recombinar informações já existentes, e não para criar conhecimento novo. E o conhecimento novo não está na internet, facilmente encontrável em um mecanismo de busca com meia dúzia de palavras-chave. Ele está por ser descoberto. E diante da complexidade da ciência e da indústria dos nossos dias, quem não souber viver em simbiose cognitiva com as máquinas (e suas redes) não terá muita chance de sobreviver. (BLIKSTEIN, 2008, não paginado).

Essa citação de Blikstein alerta para o uso das tecnologias digitais como mero veículo de acesso à informação. Por si só, isso não seria um problema, haja vista as possibilidades que esse acesso à informação por meio da internet proporciona. Porém,

²⁶ International Society for Technology in Education (Sociedade Internacional de Tecnologia na Educação).

²⁷ Computer Science Teachers Association (Associação de Professores da Ciência da Computação).

é preciso ir além do nível de usuário para criar conhecimento novo. É necessária uma transição de um papel de usuário para um papel de criador. Por conta disso, esta pesquisa enfatizou o PC – não como uma forma de pensamento superior ao raciocínio lógico e matemático, mas como uma temática potencialmente importante, tanto cultural quanto cientificamente, que emerge em meio à cultura digital.

Finalmente, destacamos que os conceitos relacionados ao PC, juntamente com os conceitos de educação, cibercultura, mediação, sociointeração, ZDP, internalização e *serious games*, compuseram os norteadores teóricos da pesquisa, os quais funcionaram como referência para a análise e a interpretação dos dados. As relações entre esses conceitos serão apresentadas e discutidas ao longo do capítulo dos resultados.

3 Método

O problema norteador desta pesquisa focou em compreender as formas pelas quais os *serious games* podem contribuir para o desenvolvimento do pensamento computacional em alunos dos anos finais do Ensino Fundamental. Para realizar tal investigação, mantendo-se a coerência com o referencial teórico, optamos pela abordagem metodológica qualitativa, pois ela “examina em *profundidade* e em *extensão* as qualidades de um fenômeno” (BRUMER et al., 2008, p. 137, ênfase adicionada).

Para Ludke e Andre (2013), a pesquisa de abordagem qualitativa caracteriza-se principalmente por utilizar do ambiente natural como fonte direta de dados e do pesquisador como seu principal instrumento. Essa característica pressupõe um convívio prolongado do pesquisador no ambiente investigado, o que, inevitavelmente, resulta na alteração desse ambiente – algo que se tornará perceptível na descrição do estudo de caso e na discussão dos resultados. Estar consciente dessa condição é imprescindível para que não se invalidem os resultados da pesquisa.

Outra característica significativa da pesquisa qualitativa é a natureza predominantemente descritiva dos dados coletados (LUDKE; ANDRE, 2013, p. 13). Geralmente, os materiais obtidos em pesquisas qualitativas consistem em descrições de pessoas, situações e acontecimentos, que costumam ser gerados a partir da transcrição de entrevistas, observações e gravações audiovisuais. Como há de se descrever mais adiante, a natureza dos dados gerados nesta pesquisa condiz com essa característica.

Nesse sentido, observamos que a pesquisa qualitativa costuma trabalhar com dados textuais, mas isso não necessariamente significa que não haja espaço para dados numéricos e estatísticos, desde que eles sejam interpretados de forma qualitativa (como ocorre neste estudo, no contexto da análise das produções dos alunos no jogo *CodeCombat*). Não obstante, como argumenta Flick:

a pesquisa qualitativa trabalha, acima de tudo, com textos. Os métodos para a coleta de informações – entrevistas ou observações – produzem dados que são transformados em textos através de gravação e de transcrição. Os métodos de interpretação partem desses textos. Diferentes roteiros conduzem em direção aos textos do centro da pesquisa, e também conduzem ao afastamento desses textos. Muito resumidamente, o processo de pesquisa qualitativa pode ser representado como sendo um caminho da teoria ao texto e outro caminho do texto de volta à teoria. A interseção desses dois caminhos é a coleta dos dados verbais ou visuais e a interpretação destes dentro de um plano específico de pesquisa. (FLICK, 2009, p. 14)

Além de explicitar a importância dos textos para a pesquisa qualitativa, Flick

descreve o processo de uma pesquisa com tal abordagem. Ao fazê-lo, Flick afirma que tal processo é composto por dois caminhos: da teoria ao texto, ou seja, do referencial teórico aos dados coletados; e do texto de volta à teoria, isto é, dos dados coletados e interpretados de volta ao referencial teórico. Ao dizer que a interseção desses caminhos consiste na coleta e interpretação dos dados, Flick se refere aos procedimentos metodológicos que possibilitam o vínculo entre a teoria e os dados gerados, os quais são fundamentais para o cumprimento dos objetivos de qualquer pesquisa.

Por fim, para demonstrar a compatibilidade desta pesquisa com a abordagem qualitativa, fazemos um vínculo entre seus objetivos específicos e as etapas do processo descrito por Flick: o primeiro caminho, da teoria ao texto, realizou-se nos dois primeiros objetivos, em que se construiu o aporte e os norteadores teóricos que fundamentariam a geração e a análise do corpus. O segundo caminho, por seu turno, é aquele que parte do texto de volta para a teoria. Esse caminho se materializou no último objetivo específico, a análise do corpus, momento em que se retornou à teoria para discussão e interpretação dos achados da pesquisa gerados no terceiro objetivo específico – a interseção entre os dois caminhos.

3.1 Procedimentos metodológicos

O objetivo geral de uma pesquisa pode determinar seu produto final, mas são seus objetivos específicos que apontam as direções em que o pesquisador deve seguir para entregar tal produto. Em outras palavras, o objetivo geral descreve o resultado que se deseja obter, enquanto os objetivos específicos adiantam as operações metodológicas (PAVIANI, 2009, p. 123-124). Nesse sentido, entendemos que é possível planejar o percurso metodológico a partir dos objetivos específicos. Por conseguinte, essa foi a forma pela qual as operações metodológicas desta pesquisa foram definidas.

O primeiro objetivo específico desta pesquisa se referia à construção de um quadro teórico que articulasse os conceitos de educação, *serious games*, pensamento computacional e a abordagem *vigotskiana*. O segundo objetivo específico, por sua vez, se referia à geração de norteadores a partir desse quadro teórico. Foram esses norteadores que indicaram os aspectos a serem observados durante a geração dos dados e, posteriormente, em sua interpretação e análise. O terceiro objetivo específico dizia respeito à constituição do *corpus* de pesquisa, isto é, aos dados gerados. Para cumprir esse objetivo, realizamos um estudo de caso, que será descrito detalhadamente na próxima subseção. O quarto – e último – objetivo específico se referia à análise do *corpus* da pesquisa. Para tanto, os dados textuais gerados foram analisados por meio de uma técnica inspirada na Análise Textual Discursiva, enquanto os códigos-fonte¹, um tipo de dado específico que aparece no caso

¹ Códigos-fonte são instruções codificadas em alguma linguagem de programação de computadores e que

estudado e que também constitui o *corpus*, foram analisados de acordo com sua adequação à solução esperada, bem como por outros fatores (como a criatividade, por exemplo). Tais procedimentos de análise serão descritos com maiores detalhes na última subseção deste item.

3.1.1 Estudo de caso

De acordo com Yin (2010, p. 39), o estudo de caso é “uma investigação empírica que *a*) investiga um fenômeno contemporâneo em profundidade e em seu contexto de vida real, especialmente quando *b*) os limites entre o fenômeno e o contexto não são claramente evidentes”. Além da visível compatibilidade com a pesquisa qualitativa, um fator decisivo na escolha do estudo de caso como método foi a questão de pesquisa.

Conforme Yin (2010), questões que procuram explicar “como” ou “por que” algum fenômeno social funciona favorecem o uso dos *estudos de caso*, dos *experimentos* ou das *pesquisas históricas*. O que diferencia os estudos de caso dos experimentos é que os experimentos exigem o controle dos eventos comportamentais e são característicos da pesquisa quantitativa, ao passo que os estudos de caso investigam os fenômenos na íntegra, em seu contexto real, de forma qualitativa. Ao mesmo tempo, o estudo de caso se diferencia da pesquisa histórica porque investiga necessariamente fenômenos contemporâneos.

Levando isso em consideração, o estudo de caso pareceu ser o método mais adequado para esta pesquisa, pois seu problema:

1. investiga de que forma os *serious games* podem contribuir para o desenvolvimento do pensamento computacional em pré-adolescentes. A expressão “de que forma” é intercambiável com “como”; logo, ela pode ser considerada apropriada para um estudo de caso;
2. tenciona investigar como um grupo de pré-adolescentes aprende e desenvolve o pensamento computacional em ambiente escolar, isto é, o ambiente natural no qual o fenômeno ocorre;
3. tenciona investigar um fenômeno contemporâneo.

A partir dessas circunstâncias, definimos que o estudo de caso seria o método pelo qual se constituiria o *corpus* de pesquisa. Na próxima subseção, descrever-se-á o caso investigado.

contém soluções parciais ou completas de problemas. Programas de computador (sistemas operacionais, editores de texto, aplicativos, jogos etc.) são criados a partir do código-fonte de inúmeros arquivos de texto.

3.1.2 Descrição do caso

O caso estudado foi constituído a partir de uma oficina de Introdução à Ciência da Computação conduzida por um professor e mediada pelo *serious game CodeCombat* (ver Apêndice A). Os participantes dessa oficina eram alunos do 7º e 8º ano do Ensino Fundamental, sujeitos cuja faixa etária parecia adequada a esta pesquisa, dado o pressuposto de que é importante fomentar o desenvolvimento do pensamento computacional desde cedo². Além disso, embora o jogo estivesse traduzido para a língua portuguesa, suas linguagens de programação apresentavam, quase que em sua totalidade, comandos escritos em inglês, o que poderia exigir dos alunos certa familiaridade com o idioma estrangeiro, algo que alunos dos anos anteriores talvez não tivessem.

Levando isso em consideração, entendemos que alunos do 7º e 8º ano, de uma escola pública, do município de Farroupilha, Rio Grande do Sul, eram os sujeitos adequados para investigar o fenômeno do desenvolvimento do pensamento computacional mediado pelos *serious games*. Os participantes da pesquisa foram designados pela diretoria da escola, que utilizou como critérios o interesse no assunto, a disponibilidade e a proximidade da residência do aluno em relação à instituição. Ao todo, a oficina contou com a participação de 10 sujeitos (ver Quadro 3).

Quadro 3 – Sujeitos da pesquisa

Sigla	Nome fictício
Ba	Bags
Ch	Cheetos
Co	Colega
Do	Donalds
Er	Ermac
Ho	Honda
Ic	Ichigo
Jo	Johan
Kn	Knight
XV	XVds

Fonte: elaborado pelo autor

No que se refere ao horário dos encontros, não deveria haver conflito entre as

² Outro fator que corroborou essa escolha foi o fato de que a desenvolvedora do jogo *CodeCombat* o recomenda para turmas de alunos a partir de 9 anos (CODECOMBAT, 2020a), ou seja, turmas do 4º ano em diante.

atividades da oficina e as atividades escolares regulares. Por isso, a oficina foi realizada aos sábados, no turno da manhã. Finalmente, a infraestrutura necessária para a realização da oficina consistiu apenas em um laboratório de informática com acesso à internet. Cabe destacar que a maioria dos computadores disponíveis para as atividades não eram capazes de rodar o jogo. Por conta disso, boa parte dos alunos organizou-se em duplas.

3.1.2.1 Estudo piloto

Por conta de incertezas como a quantidade de participantes, a quantidade de encontros e sua duração, bem como das práticas pedagógicas que seriam utilizadas, percebemos de antemão a necessidade de um estudo piloto. Além disso, um estudo preliminar foi necessário devido à inexperiência do pesquisador como professor-mediador, principalmente em turmas da faixa etária investigada. O piloto teve quatro encontros com a duração de uma hora cada (quantidade de tempo equivalente a um período de um dia letivo na instituição escolar escolhida), num contexto muito próximo daquele em que o estudo de caso seria realizado (mesma duração, período, local e infraestrutura).

Tal estudo propiciou os parâmetros necessários para a realização da oficina em que foram gerados os dados da pesquisa, tanto no que se refere à logística (cronograma, duração dos encontros, número de partícipes), quanto às práticas e ao objeto de estudo (nível de abstração das atividades em relação ao nível de desenvolvimento dos sujeitos). Com base na experiência obtida no piloto, foram removidas da “oficina oficial” atividades consideradas infrutíferas, como uma que exigia que os sujeitos planejassem suas ações antes de executá-las no jogo, bem como procedimentos de geração de dados como o questionário, que se mostrou inócuo, dado que nenhum dos participantes possuía qualquer experiência relacionada ao objeto de conhecimento, bem como o diário de progresso, que se demonstrou inviável, devido à curta duração dos encontros.

3.1.2.2 *Corpus* de pesquisa

As atividades dos alunos realizadas por meio do *serious game* foram registradas pelo próprio jogo, em uma área de acesso exclusivo ao professor. Tais registros consistem nos programas desenvolvidos pelos alunos, as soluções para os desafios do jogo, dados que constituíram uma parte substancial do *corpus*, visto que sua análise possibilitou verificar se havia processos relacionados à aprendizagem e ao desenvolvimento do pensamento computacional. Mais adiante, será explicado o procedimento que foi utilizado para analisar esse tipo de dado.

O *corpus* também foi constituído a partir da documentação das atividades por meio de anotações e de filmagens. Para realizar as anotações, o professor esteve atento aos movimentos dos alunos, aos seus questionamentos e à maneira como eles trabalhavam entre si, dentre outros elementos que podiam contribuir para a elaboração de uma resposta à

questão de pesquisa. Como as anotações por si mesmas não seriam suficientes para capturar todas as interações significativas que pudessem ocorrer na oficina, foi necessário registrar os encontros por intermédio de filmagens. Os registros audiovisuais (e suas respectivas transcrições) permitiram uma análise mais pormenorizada e aprofundada das interações entre os alunos, o *serious game* e o mediador, bem como de outros movimentos não percebidos em tempo real.

3.1.2.3 Atividades realizadas na oficina

Dentre as ações planejadas e observadas ao longo da oficina, destacamos:

- **Exploração:** buscamos sempre fomentar a imaginação e a criatividade dos alunos, incentivando-os a explorar e testar o *serious game* de todas as formas que julgassem convenientes. Os alunos mostraram-se verdadeiros exploradores, sequer aguardando por instruções para iniciar suas aventuras no jogo;
- **Exposição dialogada:** buscamos contextualizar, quando necessário, o tema estudado (ex: explicar quais ações o herói podia executar no jogo, por que elas deviam ser escritas em inglês, o que elas significavam, etc.). Para isso, expusemos o conteúdo que seria trabalhado, levando em consideração o conhecimento prévio dos alunos e os instigando a participar ativamente da exposição, como num diálogo propriamente dito. Tal prática ocorreu pelo menos em três momentos-chave da oficina:
 - no início do primeiro encontro, em que aconteceram as apresentações, as conversas iniciais e a configuração de acesso ao jogo. Nesse momento, além de conhecer um ao outro, buscamos entender as motivações dos alunos, o porquê de estarem participando de uma atividade optativa realizada nos sábados pela manhã. Reforçamos também qual era o objetivo da oficina e dialogamos a respeito do significado de programação e como ela está presente no cotidiano das pessoas;
 - no início do segundo encontro, exibimos vídeos que explicavam conceitos presentes no jogo (recursos do próprio *serious game*). Essa exposição foi necessária porque percebemos uma certa dificuldade em explicar o que eram *loops* e o conceito de variáveis, que seria abordado em seguida. Talvez porque os vídeos estavam em inglês, o engajamento dos alunos foi abaixo do esperado;
 - ao final do segundo encontro, quando reunimos os alunos em um círculo e os convidamos a explicar suas respectivas abordagens de resolução de problemas.
- **Resolução de problemas:** cada aluno interagiu com o jogo a seu modo, não sendo instruído a planejar suas ações de antemão, podendo utilizar de quaisquer estratégias para resolver os desafios, desde a exploração via tentativa e erro até a execução de um plano previamente elaborado;

- **Programação em pares:** atividade em que os alunos resolvem desafios colaborativamente. Um dos jogadores atua como “motorista”, controlando teclado e mouse, enquanto o outro atua como “navegador”, observando e planejando as ações (CODE-COMBAT, 2020b). Essa prática permitiu observarmos sujeitos de níveis diferentes de desenvolvimento interagindo entre si e, possivelmente, atuando em suas respectivas ZDP, o que foi muito valioso para a constituição do *corpus*. Devido ao número limitado de computadores capazes de rodar o *serious game*, essa foi uma das atividades predominantes durante a oficina.

3.1.2.4 Aspectos éticos da pesquisa

Dada a natureza do problema investigado, uma pesquisa com seres humanos, este trabalho foi realizado sob princípios éticos estritos. Desde o início, estava prevista a realização de todas as atividades relacionadas à geração de dados de forma estritamente ética, respeitando as resoluções nº 466/12 e 510/16 do Conselho Nacional de Saúde, que dispõem as normas éticas para a pesquisa com seres humanos.

Outro aspecto considerado foi o armazenamento dos materiais digitais produzidos ao longo da oficina. Eles foram armazenados em repositórios com acesso criptografado, o que impediu que eles fossem acessados por indivíduos não relacionados à pesquisa. Os materiais mais sensíveis, como as filmagens, serão apagados tão logo a pesquisa se encerre.

A autorização para realização da oficina foi concedida pelo representante legal da escola (ver Apêndice B). Antes do início das atividades, foi necessário realizar uma reunião com os partícipes da pesquisa, acompanhados de seus responsáveis, para comunicar e esclarecer a finalidade da oficina. Além disso, para poder participar, os sujeitos da pesquisa precisaram apresentar cópias originais, assinadas e rubricadas, dos Termos de Consentimento Livre e Esclarecido (ver Apêndice C) e Termos de Assentimento Livre e Esclarecido (ver Apêndice D); esses termos foram lidos e esclarecidos pelo pesquisador, que se disponibilizou para tirar dúvidas dos sujeitos e dos responsáveis.

No que se refere à subjetividade dos participantes da pesquisa, havia consciência de que eles eram sujeitos em fase de crescimento, cujo desenvolvimento e aprendizagem estavam em processo. Por isso, foram tomados todos os cuidados para acolher esses jovens em suas peculiaridades e especificidades. Assim, nos momentos em que eles desejaram desistir de participar ou se sentiram desconfortáveis, suas escolhas foram acolhidas (como aconteceu com os participantes Bags e Cheetos, que desistiram de participar, por exemplo). Na medida do possível, o pesquisador observou como os sujeitos lidavam com as tarefas propostas, atentando para possíveis constrangimentos, no sentido de dar orientações e fazer o que fosse preciso a fim de que eles se sentissem acolhidos em suas especificidades.

Nesse sentido, havia consciência de que as filmagens aumentariam o grau de exposição dos participantes. Por conta disso, o pesquisador esteve atento para que isso

causasse o mínimo de constrangimento e desconforto. Logo no início do primeiro encontro, os alunos foram informados e familiarizados sobre as filmagens, o que deixou alguns deles um pouco apreensivos. Contudo, ao serem informados que os vídeos não seriam divulgados, que seriam vistos apenas pelo pesquisador, passaram a agir com naturalidade, até mesmo com certa descontração. Caso eles não se sentissem à vontade por conta das gravações, as filmagens não ocorreriam, pois apesar da importância dos registros audiovisuais, seria imprescindível que os sujeitos se sentissem confortáveis e agissem de forma espontânea.

Por fim, cabe dizer que o projeto desta pesquisa foi submetido e aprovado junto ao Comitê de Ética em Pesquisa.

3.2 Procedimentos de análise

Para a análise das anotações do pesquisador e das transcrições das filmagens, utilizamos uma técnica inspirada³ na análise textual discursiva (ATD) de Moraes e Galiazzi (2011), uma abordagem qualitativa de análise de discurso, que gerou as categorias emergentes, as quais possibilitaram a interpretação dos resultados da pesquisa. As produções dos alunos no contexto do jogo, por sua vez, foram analisadas a partir de uma técnica autoral, que buscou examinar qualitativamente os registros e estatísticas de cada aluno, como por exemplo, quais conceitos do pensamento computacional foram aprendidos em quais níveis do jogo e quais conceitos foram aprendidos por quais alunos. As duas abordagens de análise serão descritas detalhadamente a seguir.

3.2.1 Análise textual discursiva

A ATD (MORAES; GALIAZZI, 2011) é uma abordagem de análise que possibilita aprofundar a compreensão de fenômenos investigados na pesquisa qualitativa. Conforme Moraes e Galiazzi (2011), a ATD é um ciclo de análise constituído por três etapas: *unitarização*, *categorização* e *comunicação*. Resumidamente, esse ciclo procura atingir novas formas de ordem por meio do caos, da desordem, ou seja, pretende desordenar e reordenar um *corpus* previamente ordenado, com o intuito de encontrar compreensões renovadas do todo.

O primeiro elemento do ciclo é a *unitarização* ou desmontagem dos textos. Após definir e delimitar o *corpus* de pesquisa, o pesquisador deve examinar seus materiais em seus pormenores, segmentando-os, de modo a encontrar unidades constituintes, também denominadas *unidades de análise* ou *unidades de registro*. Conforme Moraes e Galiazzi (2011), as unidades devem estar em consonância com os propósitos da pesquisa, o que

³ Diz-se “inspirada” na análise textual discursiva porque o *corpus* textual da pesquisa constituiu-se principalmente de transcrições de registros audiovisuais, em vez de “discursos” propriamente ditos. De qualquer forma, o método da análise textual discursiva pareceu suficientemente robusto para os propósitos desta pesquisa.

implica em um esforço de leitura e interpretação fundamentado nos norteadores teóricos, evitando assim a criação de unidades que não contribuam para a análise.

Moraes e Galiuzzi (2011) ressaltam que a definição das unidades de análise pode ocorrer em função de critérios apriorísticos ou emergentes. Nos casos em que o pesquisador conhece os “grandes temas” da análise, podem-se separar as unidades de acordo com esses temas. Ao mesmo tempo, os temas ou categorias podem ser construídos ao longo da análise, com base nos conhecimentos do pesquisador e com vistas aos objetivos da pesquisa.

Conforme mencionado anteriormente, um dos objetivos específicos desta pesquisa era gerar norteadores a partir do quadro teórico, o que indica que alguns dos “grandes temas” de análise eram conhecidos de antemão. Um esboço desses temas está disponível no Quadro 4, no qual são descritos os norteadores teóricos da pesquisa.

Quadro 4 – Norteadores teóricos

Norteador	Descritor
Mediação (social)	Estratégias do mediador (observações, problematizações, intervenções) que objetivam proporcionar acesso ao objeto de estudo.
Mediação (instrumental/tecnológica)	Acesso ao objeto de estudo por meio de instrumentos e/ou artefatos tecnológicos (um <i>serious game</i> , no caso desta pesquisa).
Internalização	Indícios de aprendizagem percebidos nas produções, questionamentos, expressões, ações e reações dos alunos.
Sociointeração	Interações com o outro; troca de experiências, de vivências, de culturas.
Interatividade	Interação explícita; participação manifesta em escolhas, eventos aleatórios, simulações etc (ZIMMERMAN, 2004)
Pensamento lógico	Aplicação dos princípios da lógica para determinar a veracidade de argumentos.
Pensamento algorítmico	Utilização do pensamento lógico para formular algoritmos.
Decomposição	Divisão de problemas complexos em partes mais simples e mais fáceis de lidar.

(continua)

(continuação)

Norteador	Descritor
Generalização e reconhecimento de padrões	Reconhecimento de padrões em diferentes problemas, de modo a generalizar uma solução aplicável para problemas similares ou um conjunto de problemas.
Abstração e modelagem	Capacidade de compreender ideias e conceitos fora de seu contexto concreto/específico e de representar tais abstrações por meio de modelos (computacionais ou não).
Avaliação	Avaliação de aspectos como exatidão, eficiência, elegância e usabilidade das soluções de um problema.

Fonte: elaborado pelo autor

Porém, isso não significa necessariamente que novos temas não emergiriam ao longo da pesquisa – e, de fato, emergiram, como se há de apresentar adiante. A análise esteve aberta para a emergência de novos temas desde o seu princípio, o que significa que houve uma abordagem mista de construção de unidades (*apriorística* e *a posteriori*).

Segundo Moraes e Galiazzi (2011, p. 195), a unitarização pode ser concretizada em três momentos distintos: “1) fragmentação dos textos e codificação de cada unidade; 2) reescrita de cada unidade de modo que assuma um significado o mais completo possível em si mesma; 3) atribuição de um nome ou título para cada unidade assim produzida”. O primeiro momento consiste em fragmentar os textos em unidades de análise e codificá-las para que seja possível manter a organização e rastrear seu documento de origem (ex: unidades 1.1 e 1.2 para o documento 1; unidades 2.1, 2.2 e 2.3 para o documento 2 etc). Antevendo esse momento, a transcrição dos vídeos foi codificada de forma metódica, com a especificação do número do encontro, da data/hora, dos nomes dos interlocutores envolvidos e de *tags*, palavras-chave que resumiam o contexto (em fragmentos considerados relevantes). No que se refere às notas do pesquisador, a documentação foi realizada em planilhas do Excel, onde se considerou os seguintes dados: sujeito a quem a nota se referia (grupo, indivíduo ou jogo), título (no caso das notas que foram geradas via Google Keep, que exigia tal informação), nota e *tags*. Por conta das facilidades oferecida pelo *software* utilizado para esse fim, a codificação foi automática (número da linha na planilha).

O segundo momento da unitarização consiste na reescrita das unidades de análise. Esse momento é necessário, pois, fora de contexto, as unidades de análise tendem a perder

seu sentido original. Conseqüentemente, deve-se reescrevê-las, de modo que apresentem com clareza os sentidos que lhes foram atribuídas. Nesse sentido, após cuidadosa leitura do contexto original em que houve a escrita, foram reescritas as unidades consideradas mais relevantes (as que possuíam *tags*), no sentido de apresentar com clareza seu sentido original.

O último momento da unitarização, que se caracteriza pela atribuição de um nome ou título para cada unidade, é uma preparação para a categorização, a segunda etapa da ATD. Além da atribuição de um título para cada unidade de texto, foi realizada uma revisão das *tags* atribuídas, no intuito de verificar sua pertinência e agrupar palavras-chave sobrepostas.

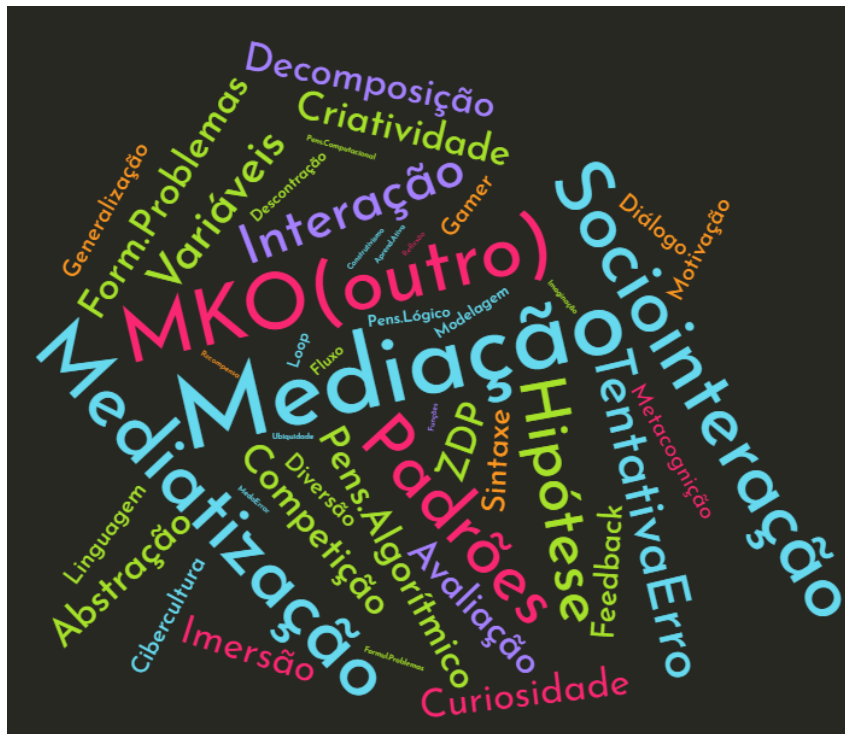
A segunda etapa do ciclo da ATD é a *categorização* ou estabelecimento de relações. A categorização é um processo que “implica construir relações entre as unidades de base, combinando-as e classificando-as no sentido de compreender como esses elementos unitários podem ser reunidos na formação de conjuntos mais complexos, as categorias” (MORAES; GALIAZZI, 2011, p. 191).

Para melhor visualizar as relações entre as unidades de análise e variáveis como frequência e relevância, utilizamos duas nuvens de palavras⁴: uma para as transcrições dos vídeos (ver Figura 1), outra para as anotações do professor (ver Figura 2). As palavras listadas nas nuvens foram geradas das *tags* atribuídas às unidades. Para chegar em seu último estado, foram realizadas várias iterações. A cada iteração, removemos palavras-chave de baixa frequência e agrupamos as que possuíam sentidos equivalentes (ex: mediação, mediações, mediação do professor). Com base na visualização das nuvens de palavras e nos norteadores teóricos, que serviram como referências, juntamente com a pergunta de pesquisa, foi possível combinar e classificar as unidades de análise em três categorias emergentes: *mediatização*, *desafio* e *mediação*.

Finalmente, chegamos à terceira e última etapa do ciclo da ATD, a comunicação. Enquanto a *primeira* etapa consistiu na fragmentação, na desorganização e na desconstrução, a *segunda* realizou um caminho inverso: gerou nova ordem e compreensão, uma nova síntese. A *terceira*, por sua vez, compreende a comunicação dessa síntese. Conforme Moraes e Galiazzi (2011), não se pretende um retorno ao texto original, mas “a construção de um novo texto, um metatexto que tem sua origem nos textos originais, expressando um olhar do pesquisador sobre os significados e sentidos percebidos nesses textos” (p. 201). Sendo assim, a última etapa da ATD consiste na comunicação desse metatexto, no sentido de explicitar a compreensão renovada do todo. No contexto desta pesquisa, tal esforço culminou em um capítulo da dissertação em que serão descritas e discutidas as categorias emergentes no sentido de elucidar as contribuições do *serious game* estudado no processo

⁴ Técnica de visualização de dados textuais em que a importância de cada palavra é definida por seu tamanho e/ou cor.

Figura 1 – Nuvens de palavras: transcrições dos vídeos



Fonte: elaborado pelo autor

Figura 2 – Nuvens de palavras: anotações do pesquisador



Fonte: elaborado pelo autor

de desenvolvimento do pensamento computacional.

3.2.2 Análise das produções do jogo

O *corpus* desta pesquisa foi constituído por materiais textuais, audiovisuais e códigos-fonte. Os materiais textuais e audiovisuais foram submetidos à ATD. Entretanto, não foi possível lançar mão da mesma técnica de análise para os códigos-fonte, formato em que consistiam as soluções dos níveis do jogo *CodeCombat* – as produções dos alunos. Logo, foi necessária uma análise distinta, que permitiu a leitura e a interpretação desse tipo de dado.

Apesar de existirem diversas técnicas de análise de código-fonte, nenhuma delas foi criada especificamente para descrever quais conceitos do pensamento computacional são empregados em cada trecho de código escrito. Desta forma, foi necessário elaborar uma técnica capaz de fazer isso. Essa técnica será descrita com maiores detalhes no próximo capítulo⁵.

O primeiro passo foi mapear quais desafios do jogo trabalhavam quais conceitos do pensamento computacional. Como esse mapeamento não existia, foi necessário observar e jogar todos os níveis do *serious game* e catalogar quais conceitos eram trabalhados em cada um deles. Para esse fim, serviram como referência os norteadores teóricos da pesquisa, principalmente os conceitos relacionados ao PC e a internalização.

Em seguida, utilizando das informações disponíveis no Painel do Professor⁶, verificamos quais níveis cada um dos alunos concluiu. A partir desses dois conjuntos de dados, foi possível realizar um cruzamento que, por sua vez, poderia dar indícios de aprendizagem.

Para realizar tal cruzamento de dados de forma mais eficiente, construímos um programa na linguagem *Python* (ver Apêndice E), que processou dois arquivos no formato CSV⁷: o primeiro continha a relação dos níveis concluídos por cada aluno (ver Apêndice F), enquanto o segundo listava a relação de conceitos por nível (ver Apêndice G). A ideia por detrás desse método era interessante, mas possuía pelo menos dois problemas:

1. todos os níveis tinham o mesmo peso, independentemente de sua complexidade e relevância dentro do *serious game*;

⁵ Optamos por apresentá-la desse modo a fim de não se revelar antecipadamente parte dos resultados.

⁶ Os códigos-fonte produzidos pelos alunos, assim como alguns indicadores de desempenho, estavam disponíveis no Painel do Professor, uma área do *serious game* restrita ao responsável pelas turmas. Esse ambiente permitiu acompanhar o progresso individual de cada aluno, fornecendo estatísticas individuais e coletivas, e abrangendo dados como o status (concluído, em andamento ou não iniciado) e o tempo levado para conclusão de cada um dos níveis, até mesmo as soluções completas elaboradas pelos alunos.

⁷ *Comma separated values* ou valores separados por vírgula.

2. apesar de ser o suficiente para dar indícios de aprendizagem dos conceitos relacionados ao pensamento computacional, isso era feito de uma forma quantitativa, binária (ou acertou, ou errou), que parecia refletir o fenômeno observado com pouca verossimilhança.

Para resolver esses problemas, ampliamos o mapeamento dos desafios do jogo, atribuindo-se pesos distintos aos níveis de maior relevância e complexidade, possibilitando assim o cálculo de uma média ponderada. Ao mesmo tempo, realizamos uma análise qualitativa das soluções dos alunos, caso a caso, nível a nível, atentando para detalhes como completude, criatividade e esforço, em vez de apenas certo ou errado⁸.

Depois disso, realizamos um novo cruzamento utilizando os novos conjuntos de dados. Novamente, construímos um programa em *Python* para realizar o processamento (ver Apêndice H). Os arquivos de entrada desse segundo programa também estavam no formato CSV e continham o mapeamento dos conceitos em relação aos desafios do jogo ponderados (ver Apêndice I) e o parecer dos alunos para cada nível (ver Apêndice J). Desta vez, o resultado pareceu refletir o fenômeno com maior verossimilhança.

Finalmente, destacamos que, em conjunto com a ATD, a análise das produções serviu como subsídio para a elaboração de uma resposta à questão de pesquisa. Obviamente, as duas técnicas de análise possuíam diferentes objetivos. Enquanto a ATD buscou uma compreensão aprofundada e renovada do *corpus*, no intuito de elucidar as “formas” em que os *serious games* contribuem para o desenvolvimento do pensamento computacional, a análise das produções dos alunos no jogo foi realizada no intuito de verificar “se” eles contribuem de alguma forma, isto é, se houve indícios de aprendizagem.

⁸ Novamente, ressaltamos que os critérios utilizados serão aprofundados durante a própria discussão e apresentação dos resultados.

4 Resultados e discussão

Possivelmente a maior contribuição desta dissertação, este capítulo apresenta e discute os resultados da pesquisa. Primeiramente, são discutidas as categorias emergentes da ATD. Em seguida, são apresentados e interpretados os resultados da análise das produções dos sujeitos no contexto do jogo. Por fim, realizamos um esforço para sintetizar os resultados observados e responder à questão norteadora desta pesquisa.

4.1 Categorias emergentes

Conforme mencionando no capítulo do método, um dos procedimentos analíticos realizados foi a ATD, que se destinou à análise das anotações do pesquisador e das transcrições das filmagens da oficina, tendo como critérios, os norteadores teóricos (ver Quadro 4) e como fio condutor, a questão de pesquisa. Inicialmente, esse método propõe que o *corpus* seja organizado em unidades de análise. Posteriormente, após uma “intensa impregnação” por parte do pesquisador, marcada por diversas idas e vindas ao texto, tais unidades são categorizadas, formando-se assim as **categorias emergentes**.

As categorias que emergiram da análise do *corpus* foram: **mediatização**, **desafio** e **mediação**. A primeira se refere principalmente à interpretação dos aspectos do *serious game* que possibilitam a aprendizagem à luz da teoria *vigotskiana*. A segunda, por sua vez, destaca os desafios do *serious game* e como eles podem desenvolver o pensamento computacional. Por fim, a terceira categoria discute aspectos relacionados à mediação do professor, que se revela importante apesar da mediatização do *serious game*.

4.1.1 Mediatização

Mediatização é um conceito da área da comunicação que costuma ser utilizado para “analisar criticamente a inter-relação entre mudanças na mídia e nas comunicações, por um lado, e mudanças na cultura e na sociedade, por outro” (COULDRY; HEPP, 2013, p. 197, tradução do Google)¹. No contexto desta pesquisa, todavia, o termo mediatização é utilizado especificamente para designar a “mediação do *serious game*”, uma mediação instrumental/tecnológica/midiática, o acesso ao objeto de conhecimento por meio de instrumentos e artefatos tecnológicos (um *serious game*, no caso desta pesquisa). Tal diferenciação se faz necessária porque adiante o conceito de mediação será abordado a partir de outro viés, o *vigotskiano*, para se referir às observações, problematizações e

¹ “[...] a concept used to analyze critically the interrelation between changes in media and communications on the one hand, and changes in culture and society on the other”.

intervenções do *outro mais experiente* (MKO), que objetivam proporcionar acesso ao objeto de estudo. Com esse esforço, esperamos eliminar qualquer ambiguidade no que tange as mediações do *serious game* (mediatização) e do professor (mediação).

A noção de mediatização aqui estabelecida engloba todos os aspectos relacionados ao *serious game* e a forma como eles interagem entre si para entregar uma experiência de jogo que possibilite ao jogador aprender. Com base no referencial teórico e no *corpus* da pesquisa, foi possível perceber como o acesso ao objeto de conhecimento por parte do jogador é mediatizado pelo *serious game*. Foram observados os seguintes aspectos: **desafio**, **interatividade**, **feedback**, **elaboração de hipóteses**, **resolução de problemas**, **aprendizagem por meio do erro (tentativa e erro e redução do medo de errar)**, **cultura dos jogos digitais e cibercultura**, e **sociointeração**. Um desses aspectos desempenha um papel tão significativo no processo de mediatização e possui tamanha riqueza de detalhes que foi promovido ao nível de categoria: o **desafio**. Por conta disso, argumentos relacionados a essa categoria serão esgotados em outra subseção, em vez desta. Os demais, por sua vez, serão apresentados e discutidos ainda nesta seção.

4.1.1.1 Interatividade: potencializando a elaboração de hipóteses e a resolução de problemas

Ao falar sobre interatividade, Zimmerman (2004) brincou que essa é uma palavra tão ambígua que pode significar ao mesmo tempo tudo ou nada. Para que isso não aconteça no contexto desta seção, cabe explicitar que a noção de interatividade utilizada é aquela que ele mesmo chama de “interatividade explícita”, que denota “interação” no sentido óbvio da palavra, a participação manifesta em “escolhas, eventos aleatórios, simulações dinâmicas e outros procedimentos programados na experiência interativa” (ZIMMERMAN, 2004, p. 158, tradução do autor)².

A partir da ATD, percebemos que essa propriedade dos jogos digitais desempenha um papel de protagonista na aprendizagem. Ritterfeld et al. (2009) corroboram ao apresentar resultados que indicam a contribuição individual da interatividade para os resultados educacionais. Os autores apontam também que:

A interatividade pode potencialmente promover o engajamento do jogador por meio da participação comportamental e do processamento cognitivo. As respostas comportamentais no jogo podem ajudar a aumentar o engajamento e a participação dos jogadores, aumentando assim o interesse do aluno e possibilitando processos de aprendizagem mais ativos. As reações instantâneas permitem ciclos de *feedback* rápidos, provocando pensamentos e aprendizados mais profundos com o engajamento do jogador no desenvolvimento do enredo por meio de diálogos, tomada constante de decisões e significação de decisões anteriores. (RITTERFELD et al., 2009, p. 692, tradução do autor)³

² “choices, random events, dynamic simulations, and other procedures programmed into the interactive experience”.

³ “Interactivity could potentially promote player engagement through both behavioral participation

Nesse sentido, enfatizamos que o *feedback* é, possivelmente, a propriedade mais importante de uma mídia interativa como um jogo digital, visto que ele proporciona uma resposta imediata para cada ação desempenhada pelo jogador, proporcionando a este a percepção de controle do estado do jogo. Tal propriedade possibilita que o jogador avalie suas hipóteses instantaneamente, o que implica em uma experiência de aprendizagem mais envolvente e motivadora (WHITTON, 2010; PRENSKY, 2003; LIEBERMAN, 2006).

A Figura 3 ilustra uma tentativa de mapear as principais situações de *feedback* que ocorrem em *CodeCombat*. As situações observadas são as seguintes:

1. *Pré-jogo*. À direita, são exibidos os objetivos e regras do desafio, normalmente em formato textual e imagem. À esquerda está o cenário em que a ação acontece.
2. *Estado inicial*. Ao clicar em “Iniciar nível”, o jogador pode dar início à resolução do problema. À direita, encontra-se o editor de código em que a solução deve ser descrita. Nele, é comum haver pistas associadas ao objetivo ou às regras do desafio, bem como algum trecho de código inicial, em que o jogador pode se basear. No centro da tela, são exibidos os comandos disponíveis ao herói no nível atual (movimentação, ataque, etc). À esquerda, no cenário, são reproduzidos os passos ordenados ao herói. Além disso, são exibidas pistas e o estado atual dos objetivos (*ex*: incompleto);
3. *Pós-erro*. Quando a solução criada resulta na morte do herói ou noutro estado que não conclua os objetivos do nível, o jogo procura informar o jogador sobre o que pode ter dado errado. Esse *feedback* costuma apontar erros de sintaxe e erros de lógica, além de mostrar qual dos objetivos não foi atingido;
4. *Pós-resolução do problema*. Quando a solução elaborada pelo jogador está correta, o jogo informa que todos os objetivos foram concluídos e permite o avanço para o próximo nível.

A partir da elucidação dessas situações, podemos perceber que um ciclo se repete durante a interação dos jogadores com o jogo: de início, buscam compreender o problema a partir dos objetivos, regras, pistas e *feedback* prévio; em seguida, elaboram uma hipótese para resolver tal problema e escrevem um algoritmo que a descreva; por último, a partir da observação da execução do programa, verificam se a hipótese elaborada se sustenta, isto é, se ela está correta e resolve o desafio proposto. Se a solução estiver correta, pode-se partir para o próximo desafio. Por outro lado, se a solução estiver errada, o jogador precisa passar pelos mesmos passos; uma nova iteração ocorre. Logo, trata-se de um ciclo, o **Ciclo**

and cognitive processing. The behavioral responses in game could help enhance player involvement and participation, thus boosting learner interest and enabling more active learning processes. Instant reactions allow for quick feedback loops to provoke deeper thinking and learning with player engagement in the plot development through dialogues, constant decision-making, and sense-making of previous decisions”.

Figura 3 – Situações de *feedback* em *CodeCombat*

Fonte: elaborado pelo autor

de Mediatização Interativa (CMI)⁴. Indícios desse ciclo são observáveis em um diálogo entre **Ichigo** e **Colega**, ocorrido no segundo encontro da oficina:

ICHIGO : Ó os espinhos!

COLEGA : É só fazer um *loop* infinito!

ICHIGO : Não, não.

COLEGA : Claro! Desce...

ICHIGO : Dependendo do *loop* infinito a gente pode pegar nos espinhos!

COLEGA : Não, ele não pega. Só se ele ficasse... (*convence* ICHIGO)

ICHIGO : Éeee.

COLEGA : Só isso aí.

ICHIGO : E se bugar o jogo também. [...] Olha só os movimentos que ele tá fazendo!

COLEGA : Pronto!

ICHIGO : Tu viu como ele ficou girando ao redor de si mesmo?

COLEGA : Ele fez uma coisa errada.

ICHIGO : Ah, não me diga!!

COLEGA : Ah o cara, né! Tá, vamo vê, vamo vê. (*exploram um pouco a interface do jogo e percebem que há uma lista de comandos disponíveis para o herói*)

COLEGA : Ah, aqui tem os comandos, ó!

ICHIGO : Quê?! Quer dizer que esses aí é todos os comandos que a gente podia usar?

COLEGA : [...] Ele [herói] teria que vim aqui (*aponta*). Daí, se fosse *loop* infinito, ele ia ficar assim, ó: (*gestualiza*). Aqui que ele morre.

ICHIGO : Sim.

COLEGA : Então ele teria que...

ICHIGO : Mas ele não precisa ir aqui, ele não precisa...

COLEGA : Quando ele desce aqui, ele tem que ir 2 pra frente e sobe de novo. Então é: *moveRight* duas vez..

ICHIGO : Já tem o *moveRight* aqui ó (*na lista de comandos*)

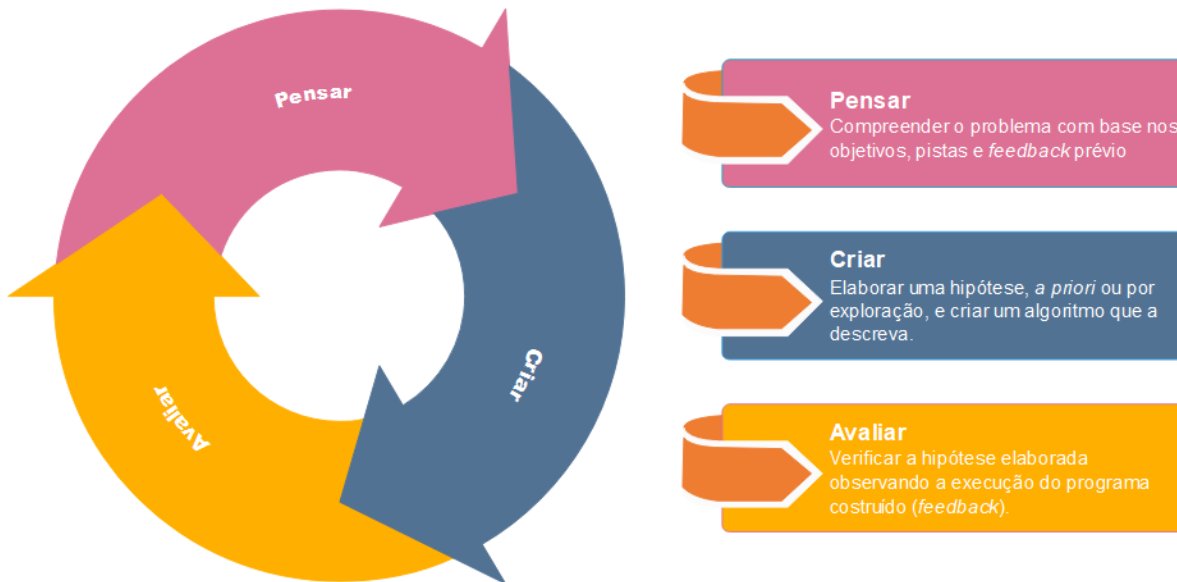
COLEGA : Sim, mas ele só vai uma vez então ... (*ele já havia escrito, por isso sabia*). E daí ele sobe.

ICHIGO : Duas, infinitamente!

⁴ Olhando em retrospectiva, é inegável que exista certa semelhança entre o Ciclo de Mediatização Interativa e as etapas de resolução de problemas descritas por Polya (1971): compreensão, planejamento, implementação e avaliação.

Figura 4 – Ciclo de Mediatização Interativa

Ciclo de Mediatização Interativa



Fonte: elaborado pelo autor

COLEGA : Uma. Vamo agora rodar. Pronto, ganhamo!

Esse diálogo apresenta parte da trajetória de uma dupla de alunos em um dado nível de *CodeCombat*. De maneira isolada, pode ser difícil compreender o que se passava em sua mente, o que demonstra o quão imersos no jogo eles estavam. Porém, é possível compreender que estavam aprimorando suas hipóteses, refinando-as a partir do *feedback* do jogo e das pistas que ele oferecia. Ao final, percebemos o momento da verificação da resposta, concluindo assim, o CMI.

O CMI é dividido em pelo menos três etapas, as quais abarcam a resolução de problemas: pensar, criar e avaliar. Na oficina, de modo geral, os alunos buscavam compreender os problemas antes de tentar resolvê-los, solicitando a ajuda dos colegas ou do professor quando necessário. Não obstante, em certos momentos, alguns deles começavam a escrever a solução antes de conhecer os objetivos do nível em que estavam ou sem prestar atenção às dicas fornecidas pelo jogo (como se percebe no trecho de diálogo acima, entre **Ichigo** e **Colega**) – como se não reservassem o tempo necessário para **pensar**. Consequentemente, não era raro que falhassem, pois acabavam agindo antes de ter os subsídios necessários para se chegar a uma solução correta (tais subsídios são apresentados pelo jogo de forma destacada, como os objetivos, pistas e novas regras).

No que se refere ao **criar**, a interatividade do jogo possibilita que cada sujeito organize seu processo criativo da maneira que julgar conveniente, como se pode observar

em uma conversa do professor com os alunos:

PROFESSOR : Então, aqui uma coisa que talvez vocês tenham percebido: às vezes não é uma boa ideia simplesmente entrar ali no jogo e começar a botar um monte de comandos sem pensar no que vocês têm que fazer antes. Então, tipo, vocês vão ali [no jogo], tentam identificar o objetivo que vocês têm que cumprir e depois vocês começam a fazer um plano. Vocês têm trabalhado mais ou menos assim? Como é que vocês têm trabalhado? Ou vocês lembram de ter trabalhado semana passada [retrasada]?

BAGS : Pensando, né sor! Eu “trabalho” pensando no que tem que fazer.

KNIGHT : Ah, eu olho o objetivo e tipo: “ah, tem que ir pra direita e subir”, depois olho os códigos [exemplos] e [...] pra programar.

PROFESSOR : E tu tenta, tipo, pensar e identificar a solução antes de começar ou tu vai tipo, ah, faz um pedacinho dela e depois faz outro pedacinho...

KNIGHT : Penso tudo antes de começar.

PROFESSOR : Tu tenta pensar todo o caminho que tu faria antes de começar?

DONALDS : Eu tento pensar todo o caminho e coloco nos códigos e se eu não consigo, eu vou mudando até eu conseguir.

PROFESSOR : Vocês fazem mais no estilo que o DONALDS falou ou o KNIGHT, mais ou menos assim? Tentam pensar tudo de uma vez só ou...

BAGS : Depende do que tem que fazer, né sor. Se é difícil assim, o cara vai fazendo por partezinha.

A partir dessa transcrição, percebemos que, enquanto alguns alunos – como **Knigh**t – planejavam seus passos antes de agir, formulando hipóteses *a priori* (escreviam todo o programa e só então o executavam), outros construíam e reconstruíam suas hipóteses por meio da exploração (tentativa e erro), com base no *feedback* contínuo do jogo (descreviam um programa que resolvia apenas parte do problema, então executavam-no, verificavam o que houvera de errado, e realizavam as alterações necessárias, sucessivamente).

Outrossim, a partir do *feedback* do *serious game*, o aluno é capaz de **avaliar** suas próprias hipóteses. Dessa questão, há pelo menos duas decorrências:

1. o jogador é instigado a exercitar o princípio da *avaliação*, um dos conceitos relacionados ao pensamento computacional;
2. ele não precisa aguardar uma correção do professor, visto que o jogo se encarrega dessa questão⁵.

⁵ Contudo, ressalta-se que tal característica não implica em abrir mão de um professor-mediador; muito

Finalmente, defendemos que os aspectos relacionados à interatividade – como o *feedback* instantâneo – possibilitam a verificação de hipóteses e a resolução de problemas de uma forma rápida e dinâmica, o que potencializa a aprendizagem da linguagem e dos desafios do jogo por meio da exploração. Tais aspectos constituem o CMI, que desempenha um papel crucial no desenvolvimento do pensamento computacional, dada a natureza dos desafios de *CodeCombat*.

4.1.1.2 Aprendendo com os erros

Se existe a possibilidade de que algum provérbio seja conhecido por todos, este seria “errar é humano”. Não obstante, na prática, é comum que se perceba o fracasso como algo inadmissível, seja no ambiente de trabalho, de estudo ou até mesmo de lazer. No contexto da aprendizagem escolar, essa percepção tende a aumentar o medo de errar, podendo restringir a curiosidade, um fator importante na aprendizagem (MALONE, 1981).

Conforme argumenta o matemático Marcus (2017, p. 8, tradução do autor)⁶, “dado que as falhas são inevitáveis, é essencial ensinar os alunos a resistir ao fracasso, bem como entender que o fracasso é normal. Além disso, sempre há algo a aprender com uma falha”. Nesse sentido, parece razoável afirmar que os jogos digitais são apropriados para tal finalidade, pois ao jogá-los o jogador não precisa se preocupar com seus erros: ele pode tentar de novo, pois há infinitas vidas e chances de aprender (MICHAEL; CHEN, 2005; YONG; GATES; CHAN, 2019).

Desta forma, o ambiente seguro proporcionado por um jogo digital como *CodeCombat* abre uma nova gama de possibilidades e caminhos para o conhecer, visto que nele o sujeito pode falhar sem as consequências do mundo real, permitindo que ele não restrinja mais sua criatividade, libertando sua imaginação para buscar novos ângulos de resolução de problemas.

O professor-mediador também pode se valer dessa noção para organizar sua abordagem em sala de aula – algo que ocorreu na oficina. Baseado no pressuposto de que sempre há algo a aprender com o fracasso, em determinados momentos o professor sugeriu aos alunos que realizassem uma ação no jogo que provavelmente os conduziria a um erro, como nesse diálogo entre ele, **Bags** e **XVds**, a respeito do nível ‘O Prisioneiro’:

PROFESSOR : já libertou o arqueiro pra ele te ajudar?

XVDS : sim, mas ele não ajuda!

pelo contrário. Como se há de argumentar mais adiante, o professor pode utilizar esse tempo extra para problematizar, dar pistas e propor outras atividades.

⁶ “Given that failures are inevitable, it is essential to teach students to become resistant to failure, as well as an understanding that failure is normal. Moreover, there is always something to learn from a failure”

PROFESSOR : é que tem uma questão assim, ó: ele meio que te segue, né? Então, pensa assim: um arqueiro, se ele fosse lutar contra um cara de espada, à curta-distância, ele conseguiria matar o cara?

BAGS

XVDS : ... [acenam que não]

PROFESSOR : Então tem que fazer o seguinte: tenta atrair a atenção do guerreiro, e deixa o arqueiro livre pra dar flechada. Tenta fazer isso. De repente, nem ataca ele [o inimigo], vai pro canto e vê o que acontece.

Em situações como essa, não somente pode o aluno perceber que sua solução está errada, mas também entender o porquê disso. Entender porque uma solução não funciona é estar mais próximo de uma solução que funcione, assim como é dar um passo na direção da internalização (principalmente de conceitos como o *reconhecimento de padrões*).

Apesar disso, nem todos os sujeitos da pesquisa se mostraram confortáveis com o erro. Enquanto alguns não demonstravam medo de errar sob nenhuma circunstância, outros se retraíam um pouco quando tinham dúvidas – talvez por medo da represália de seus pares –, principalmente no primeiro encontro da oficina. Ademais, em vez de considerar o fracasso como uma oportunidade de aprender, alguns alunos se frustravam quando erravam. Em determinados momentos, foi possível notar como **Ichigo** e **Honda** perderam um pouco da motivação depois de errar sucessivamente (Ichigo costumava irritar-se com o jogo, ao passo que Honda se dispersava da atividade). Por outro lado, com o passar dos encontros, mesmo os sujeitos mais retraídos começaram a solicitar ajuda do professor ou dos colegas quando se deparavam diante de um desafio que não conseguiam resolver.

De qualquer modo, podemos perceber a importância dos erros em *CodeCombat* e nos jogos digitais em geral. Em virtude de inúmeras tentativas malsucedidas durante a oficina, os jogadores precisaram concentrar-se e perseverar para aprender as mecânicas e dinâmicas do jogo, como a sintaxe, as funções, os algoritmos, os *loops* etc. Para tanto, passaram por vários CMI, isto é, identificaram problemas, desenvolveram uma interpretação causal de eventos, conjecturaram e implementaram possíveis soluções, e avaliaram resultados. Assim, convém concordar com Squire (2005, tradução nossa)⁷: “o fracasso não é apenas um ‘problema’, mas também uma condição prévia crítica para a aprendizagem”.

4.1.1.3 Cultura dos jogos digitais e cibercultura

Conforme discutido no referencial teórico, as TDIC têm mudado a forma de viver e de se relacionar das pessoas, inclusive no que toca à educação. A experiência vivenciada na oficina mediatizada pelo *serious game CodeCombat* corrobora essa questão: a escola

⁷ “[...] failure was not only a "problem" but also a critical precondition for learning”

já não é mais como há 15 anos, quando poucos alunos costumavam acessar a Internet e jogar jogos digitais com frequência. Atualmente, a maioria (senão todos) os alunos jogam, diariamente, tanto em computadores *desktop* e *videogames* quanto em *smartphones*. O acesso à internet é pervasivo; tornou-se uma necessidade.

Nesse sentido, conforme esperado, a introdução de *CodeCombat* no contexto de uma escola pública não trouxe espanto aos alunos; pelo contrário, estes sentiram-se muito confortáveis com a ideia de ir à escola nas manhãs de sábado para jogar e aprender um pouco sobre programação. Isso não ficou evidente apenas na motivação e no engajamento dos alunos, mas também em suas formas de agir e de se comunicar. A fluência com que utilizavam a linguagem do mundo dos jogos ficou aparente em diversos momentos da oficina, como por exemplo uma situação em que o professor pediu a **KnighT** que explicasse o que havia feito para derrotar sua estratégia no nível *Wakka Maul*:

PROFESSOR : (*retorna a atenção à KNIGHT; senta-se ao lado dele*) Como é que tu fez pra derrotar o meu carinha?

KNIGHT : Eu não sei... [...] e eu morri (*risos*).

KNIGHT : Eu só libertei essa porta que eu não tinha libertado antes, aí...

PROFESSOR : Ah sim, sim!

KNIGHT : Eu ia subir até lá e *summonar* os carinhas pra te matar, mas aí quando eu abri essa [porta] daqui ele *summonou* sozinho e matou.

Nesse excerto da transcrição das gravações da oficina, uma frase específica chama a atenção por estar permeada de termos comuns ao mundo dos *games*: “*summonar* os carinhas pra te matar”. “*Summonar*” é um aportuguesamento da palavra *summon*, da língua inglesa, que significa convocar. “Carinha”, por sua vez, é um termo bastante comum para se referir ao conceito de personagem. Também cabe ressaltar a naturalidade dos dois ao falar sobre matar um ao outro: no contexto desse diálogo, não há nada de errado em matar um adversário, pois está subentendido que as ações realizadas afetam personagens virtuais do mundo do jogo. É um atalho discursivo para simplificar o diálogo. No final das contas, tal frase pode ser interpretada como “convocar os meus guerreiros pra derrotar teu personagem”.

À primeira vista, argumentar que detalhes linguísticos como esse contribuem para a mediatização pode parecer exagero. Não obstante, um olhar mais atento perceberá que não se trata de exagero, mas sim de coerência com a noção de aprendizagem *vigotskiana*, visto que aprender e compreender os signos de uma atividade são os primeiros passos para internalizá-la (VIGOTSKI, 2007).

Outra questão relacionada à cibercultura que contribui para a mediatização de um *serious game* é a ubiquidade das TDICs. Como um artefato tecnológico disponível no

ciberespaço, *CodeCombat* pode ser acessado de qualquer lugar, a qualquer momento, seja no contexto de sala de aula ou no conforto de uma sala de estar. Tal característica não deve ser desprezada, visto que os alunos que se interessarem podem continuar a jogar e aprofundar seus estudos por conta própria, não se limitando ao período de aula.

Para corroborar essa conjectura podemos mencionar um evento ocorrido entre o primeiro e o segundo encontro da oficina. Um dos sujeitos da pesquisa, **Donalds**, não pôde estar presente no primeiro dia. Essa falta poderia afetar diretamente seu aproveitamento nos demais encontros, visto que nessa ocasião foi apresentada a proposta da oficina e iniciada a campanha no jogo. Não obstante, isso não aconteceu, pois Donalds demonstrou interesse ao ser-lhe apresentada a possibilidade de acessar e explorar o *serious game* de sua casa. O convite para participar à distância e fora do período da oficina foi enviado por meio de um aplicativo de conversas instantâneas⁸, assim como as informações de acesso do jogo. Seguindo seu próprio ritmo, Donalds concluiu boa parte dos níveis ainda na primeira semana, tendo chegado ao segundo encontro quase no final do jogo – sem qualquer intervenção do professor, o que demonstra a capacidade de mediatização de *CodeCombat*.

Assim, percebemos que as características ciberculturais de *CodeCombat* (um jogo online e desterritorializado) e as circunstâncias socio-histórico-culturais em que ele é utilizado (como a cibercultura e a linguagem) também contribuem para a mediatização da aprendizagem do pensamento computacional.

4.1.1.4 Relações entre mediatização, sociointeração e internalização

Os processos psicológicos superiores como o pensamento computacional “surgem a partir do desenvolvimento cultural da criança, cuja fonte são a colaboração e a aprendizagem” (VIGOTSKI, 2009, p. 335). Em outras palavras, tais processos são decorrentes da *internalização* e da *sociointeração*.

Conforme Vigotski (2007, p. 56), internalização é “a reconstrução interna de uma operação externa”, isto é, a transformação de um processo interpessoal (social) em um processo intrapessoal (individual). O autor argumenta também que “a internalização de formas culturais de comportamento envolve a reconstrução da atividade psicológica tendo como base as operações com signos” (VIGOTSKI, 2007, p. 58). Deduzimos, portanto, que o “primeiro momento” do processo de internalização envolve a aquisição de signos desenvolvidos cultural e historicamente. No aporte teórico do autor, observamos que tal aquisição ocorre por meio da interação com o **outro**, o alguém mais experiente (MKO). Nesta pesquisa, entretanto, propomos uma ampliação de tal noção ao defendermos que, por meio da mediatização, *serious games* também possam participar no processo de internalização, atuando em conjunto ou até mesmo no lugar do MKO. Por conta disso,

⁸ O aluno estava devidamente inscrito na oficina e, por conta disso, havia ingressado no grupo de *WhatsApp* da turma.

optamos por acoplar e discutir o conceito de sociointeração no contexto da *mediatização*, ao invés da mediação.

Nessa perspectiva, um *serious game* deve fornecer os signos necessários para que o pensamento computacional – a atividade psicológica estudada – seja reconstruído internamente pelo jogador. Com base no que se observou na oficina, defendemos que o jogo *CodeCombat* não somente provê os instrumentos e sistemas simbólicos para que seja possível resolver os desafios propostos, como também explica as relações entre os signos e as formas de utilizá-los. *CodeCombat* fornece diversas pistas, por meio de diferentes sistemas simbólicos (imagens, texto, código-fonte e vídeos), para que o jogador tenha à sua disposição as informações mínimas (e às vezes muito além disso) para aprender a linguagem de jogo e elaborar hipóteses corretas para os problemas que se lhe apresentam. Na medida em que o sujeito assimila tais instrumentos psicológicos, ele se torna capaz de realizar as mesmas operações mentais sem que o referencial concreto esteja presente, isto é, os signos adquiridos por ele não servem apenas para resolver questões que se apresentam em seu campo visual imediato, mas também para o planejamento de ações futuras (VIGOTSKI, 2007, p. 14).

No entanto, por mais que um jogo forneça os signos necessários aos jogadores (regras, palavras-chave, comandos, símbolos, ícones), possibilitando que estes joguem e aprendam por conta própria, é importante que ele também incentive interações pedagógicas entre os pares – a sociointeração⁹. Sob a perspectiva *vigotskiana*, a sociointeração – sobretudo a colaboração – exerce um papel importante, uma vez que “a criança orientada, ajudada e em colaboração sempre pode fazer mais e resolver tarefas mais difíceis do que quando sozinha” (VIGOTSKI, 2009, p. 328). Isso ocorre principalmente porque o MKO pode gerar as condições necessárias para que um sujeito consiga realizar algo que não conseguiria fazer sozinho, seja por meio de problematizações e pistas, seja por meio da imitação. Uma simples explicação por parte do MKO pode ser o suficiente para que seu interlocutor consiga resolver um problema que não conseguiria por conta própria. Ao mesmo tempo, o sujeito aprendiz pode realizar aquilo que não conseguiria sozinho por meio da imitação. Contudo, “ao contrário do senso comum, não é possível imitar qualquer coisa. Uma pessoa só consegue imitar aquilo que está ao alcance de suas possibilidades, de seu desenvolvimento” (VIGOTSKI, 2009, p. 328), ou seja, aquilo que está ao alcance de sua ZDP. Portanto, é importante que o MKO tenha a capacidade de discernir o nível de desenvolvimento de seu interlocutor. Esse argumento será retomado mais adiante, na seção que se refere à mediação do professor.

De qualquer modo, destacamos que as interações pedagógicas entre os alunos durante a oficina mediatizada pelo jogo *CodeCombat* foram muito ricas (ver Figura 5),

⁹ O termo é bastante empregado na pesquisa brasileira relacionada a Vigotski. Por outro lado, não é comum encontrá-lo na literatura disponível em inglês, por exemplo. Em artigos escritos nesse idioma, a expressão “*social interaction*” costuma adquirir o mesmo sentido de sociointeração.

Figura 5 – Sociointeração e mediação do professor em um encontro da oficina



Fonte: elaborado pelo autor

apesar de ser um jogo *singleplayer* (único jogador). Por outro lado, é possível que tais interações tenham ocorrido devido à abordagem *vigotskiana* do professor-mediador, que encorajava a colaboração entre os alunos, e das circunstâncias do laboratório disponível para a realização da oficina, que não possuía computadores suficientes para cada aluno.

A sociointeração possibilita a troca de experiências entre pares com níveis de desenvolvimento intelectual diferentes e, conseqüentemente, propicia a mobilização da ZDP (VIGOTSKI, 2007; DJAOUTI, 2016), uma vez que os alunos mais experientes podem ajudar seus colegas a ir além daquilo que conseguem fazer sozinhos. Ao longo da oficina, pelo menos dois alunos se destacaram e demonstraram capacidade de atuar como MKOs: **Knight** e **Donalds**. Em alguns momentos, os dois atuaram mais como MKOs, fornecendo pistas e problematizando em colaboração com seus pares, do que como jogadores *per se*. No diálogo abaixo, por exemplo, Donalds auxilia o professor a explicar um determinado conceito a Knight:

KNIGHT : O que que é “argumentos”? ... O que são argumentos?

PROFESSOR : Argumento é tipo o que tu bota dentro de um poder ali (*explicação rasa, mas contextualizada*). Quando tu faz assim, tipo: “moveRight”, tu bota “2”, pra andar duas vezes, isso é um argumento. Ele [o jogo] tá reclamando de algum erro?

KNIGHT : Aqui tá dizendo que eu tenho que usar “argumento”, “variável” e “loop”.

DONALDS : Ah.. Eu tô nessa fase aí também.

PROFESSOR : Quer dizer que tu tem que usar alguma função. Então, tipo, em vez de usar “moveRight”, tem que usar “moveRight(2)”, por exemplo.

DONALDS : Tu [KNIGHT] tem que usar, tipo, “While-True Loop”, o “moveDown”, o “moveUp” e o ataque.

Ademais, a atuação de Donalds como MKO ficou evidente no segundo encontro da oficina, quando assumiu a responsabilidade de guiar **Cheetos**, aluno que faltara ao primeiro encontro. Em dado momento, Donalds solicitou a ajuda do professor para conseguir explicar a Cheetos o que era um *loop*, o que demonstra um senso de responsabilidade para com a aprendizagem de seu colega:

DONALDS : Ô sor! (PROFESSOR *se aproxima*) Tu consegue explicar pra ele [CHEETOS] o que é o *Loop*? Eu sei usar, mas não sei explicar.

A perceptível fluência na linguagem e a facilidade em lidar com os desafios do jogo apresentada por Knight e Donalds, assim como por **Ermac** e pelos demais em alguns momentos, dá indícios da reconstrução interna de comportamentos aprendidos socialmente, o “segundo momento” do processo recursivo que é a internalização. A reconstrução interna inicia-se a partir do momento em que os signos apreendidos no contexto do jogo passam a fazer parte do processo de pensamento do aluno, quando ele passa a construir mentalmente utilizando-se de conceitos como algoritmos, instruções, *loops* e variáveis, ou até mesmo comandos específicos (*moveRight* e *while*, por exemplo). Além disso, a internalização se evidencia quando os sujeitos começam a se comunicar com seus pares utilizando tais signos, como ocorreu em diversas situações ao longo da oficina:

JERRY : Ihhh.. Agora sim! Agora vai!

COLEGA : Menos de 7 declarações! Tem umas quinze! (*risos*). Tem que ser 7. ... Vamo colocar *loop*!

COLEGA : *Left*. Daí, *Up*. *Up*.

JERRY : E agora... 2.

COLEGA : 2!

JERRY : E agora?

COLEGA : Bota um *loop*.

...

COLEGA : (*chama o professor para pedir ajuda*)

PROFESSOR : (*olha a tela*) Ah tá... É que... aqui ó: (*mostra na tela*) Ele tá dizendo aqui, ó (*aponta*). Senta aí, ICHIGO. Quer dizer, pode sentar, ICHIGO. Não foi uma ordem. Desculpa. Se tu quiser, tu pode sentar.

ICHIGO : Tem que botar os código do comando pra fazer sentar, sor! (*risos*)

PROFESSOR : ICHIGO.*sit()*

Finalmente, ressaltamos que a mediatização do *serious game CodeCombat* potencializa o processo de internalização, tanto por facilitar a aquisição dos signos necessários para a realização de atividades relacionadas ao pensamento computacional, atuando quase como um MKO, quanto por encorajar a sociointeração, que permite a aprendizagem por meio da imitação, mobiliza a ZDP e ensina a colaborar com o outro.

4.1.2 Desafio

Os aspectos de mediatização do *serious game* discutidos na seção anterior são importantíssimos para a aprendizagem; sem eles, podemos arriscar dizer que o jogo *CodeCombat* apresentaria poucas possibilidades de mediatizar o acesso ao pensamento computacional. Não obstante, por si mesmos, tais aspectos pouco – ou nada – agregam para o desenvolvimento de tal forma de pensamento. Isso pode soar contraproducente, mas a tecnologia por si só, por maior que seja seu potencial, não garante a aprendizagem, independentemente de qual seja o objeto de estudo – e isso também se aplica aos jogos sérios. Um *serious game* desprovido de conteúdo voltado à aprendizagem de um dado assunto é senão um jogo vazio. Por outro lado, um *serious game* com grande potencial de mediatização aliado a um conteúdo embasado em teorias de aprendizagem, com uma ementa bem-definida, carrega em si grandes possibilidades de aprendizagem.

Nesse sentido, cabe ressaltar que esse foi o motivo pelo qual *CodeCombat* foi escolhido dentre tantos outros jogos: seu *game design* foi concebido e elaborado com base num objeto de conhecimento específico, o pensamento computacional. Por conta disso, seus desafios buscam, a cada instante, mediar o acesso ao pensamento algorítmico e lógico, à decomposição, à generalização, ao reconhecimento de padrões, à abstração, à modelagem e à avaliação.

Para corroborar essa afirmação, ao longo desta seção, serão apresentados e discutidos os desafios do jogo que mais parecem contribuir para o desenvolvimento do pensamento computacional.

4.1.2.1 Critérios de escolha dos desafios analisados

Desde o primeiro contato com o jogo, houve a impressão de que, levando-se as definições de cada conceito ao limite, quase *todos* os desafios de *CodeCombat* empregavam *todos* os conceitos centrais do pensamento computacional. Em outras palavras, pensamento lógico, pensamento algorítmico, decomposição, generalização e reconhecimento de padrões, modelagem, abstração e avaliação estão, de forma latente ou manifesta, contidos nos níveis do *serious game* pesquisado. Para ilustrar esse ponto, cabe observar o primeiro nível do jogo, “Masmorra Kithgard” (ver Figura 6).

Por ser o primeiro nível do jogo, “Masmorra Kithgard” apresenta um desafio

Figura 6 – *Screenshot* do Nível 1 - “Masmorra Kithgard”, durante a apresentação dos objetivos



Fonte: elaborado pelo autor

bastante simples: “colete a gema e escape da masmorra”. Para abordar esse problema, por menor que ele seja, o jogador pode subdividi-lo em: movimentação e coleta da gema (*decomposição*). Para solucioná-lo, o jogador precisa desenvolver uma linha de raciocínio para chegar a uma conclusão (*pensamento lógico*) e descrevê-la, passo-a-passo, de forma não ambígua (*pensamento algorítmico*). Tal descrição precisa ser compreendida pelo computador; logo, deve ser descrita em linguagem de programação, em vez da linguagem natural. Assim, a variável `hero` passa a representar o herói do jogo, ao passo que `moveRight` e `moveDown` são representações abstratas de comandos de movimentação (*abstração e modelagem*). Após descrever uma possível solução (hipótese), o jogador precisa executá-la para testá-la, isto é, verificar se ela está correta (avaliação). Caso não esteja, deverá reiniciar o ciclo, descrevendo novas hipóteses e testando-as, até que uma delas esteja correta.

Assim, percebemos como até mesmo o mais simples dos desafios de *CodeCombat* emprega a maioria dos conceitos base do pensamento computacional. Os conceitos de reconhecimento de padrões e generalização, que não aparecem diretamente no primeiro nível, estão ausentes por causa de suas características intrínsecas, que se evidenciam melhor na perspectiva nível a nível, na observação das transições de um nível a outro (ou em problemas específicos de reconhecimento de padrão, como os que envolvem *loops*, por exemplo).

O ponto ao qual se quer chegar é que não é necessário discutir cada um dos conceitos em cada um dos níveis de *CodeCombat*, pois, em última instância, todos os níveis trabalham todos os conceitos do pensamento computacional. Tal esforço resultaria em uma

```
1 hero.moveRight()  
2 hero.moveDown()  
3 hero.moveUp(2)  
4 hero.moveRight()
```

Código 1 – Solução do autor para o nível “Gemas nas profundezas”

discussão desnecessariamente extensa. Ao invés disso, o foco da discussão desta parte da análise serão os níveis que se destacam dos demais na aplicação de um ou mais conceitos.

4.1.2.2 Masmorra Kithgard: introdução ao pensamento computacional

Esse é o primeiro nível do jogo e, por conta disso, tem um dos objetivos mais simples: evitar os espinhos e coletar a gema. Por ter sido descrito em detalhes anteriormente, não é necessário esmiuçá-lo mais uma vez. Não obstante, vale ressaltar que ele traz em si o primeiro contato com os conceitos de abstração e modelagem, mesmo que de forma implícita e transparente ao jogador: a variável `hero` representa o herói do jogo, aquele que realiza as ações; os métodos (comandos) `moveRight` e `moveDown`, por sua vez, representam quais ações de movimentação são possíveis a esse herói. A modelagem está presente na variável e nos métodos, ambos modelos abstratos de objetos que existem no contexto do jogo, ao passo que a abstração está latente na compreensão de tais modelos, na capacidade de compreender que o controle das ações do herói se dá por meio de instruções envolvendo a variável `hero`, em vez do controle direto do herói através de botões.

4.1.2.3 Gemas nas Profundezas: pensamento algorítmico e argumentos

Em Gemas nas Profundezas, segundo nível do jogo, encontramos o primeiro desafio em que há mais de um caminho disponível (ver Figura 7). Para completá-lo, o jogador poderia, por exemplo, ir à direita, para baixo, para cima duas vezes e à direita (ver Código 1). Ao invés disso, também poderia ir à direita, para cima, à direita, à esquerda e para baixo duas vezes (ver Código 2).

Em situações como essa, em que há possibilidade de traçar múltiplos caminhos, o jogo convida o jogador a exercitar seu pensamento algorítmico, pois este passa a perceber como a ordem dos comandos realmente importa, não somente em termos de eficácia (se os objetivos são cumpridos), mas também no que se refere à eficiência (se a solução elaborada é ótima, isto é, utiliza o melhor caminho possível, com o menor número de instruções). Isso pode ser percebido em uma comparação entre as soluções descritas nos Trechos de Código 1 e 2: o primeiro é mais eficiente que o segundo, pois utiliza quatro instruções, enquanto este utiliza cinco.

Figura 7 – Screenshot do Nível 2 - “Gemas nas Profundezas”



Fonte: elaborado pelo autor

```
1 hero.moveRight()  
2 hero.moveUp()  
3 hero.moveRight()  
4 hero.moveLeft()  
5 hero.moveDown(2)
```

Código 2 – Solução sugerida pelo jogo para o nível “Gemas nas profundezas”

Ainda no que toca à eficiência e, conseqüentemente, ao pensamento algorítmico, este nível de *CodeCombat* introduz o conceito de argumentos, valores passados à uma função ou método, com intuito de modificar seu comportamento. No contexto do jogo, os primeiros métodos que podem receber argumentos são os de movimentação (*moveUp*, *moveDown*, *moveRight* e *moveLeft*). Por padrão, esses métodos fazem o herói dar um passo na direção escolhida. O jogador pode mudar esse comportamento passando como argumento o número de passos desejados. Conseqüentemente, ele é capaz de otimizar suas soluções, reduzindo o número de instruções necessárias (em vez de ordenar ao herói *moveUp* n vezes, ele pode obter o mesmo resultado utilizando *moveUp(N)*).

Figura 8 – *Screenshot* do Nível 3 - “Guarda sombrio”

Fonte: elaborado pelo autor

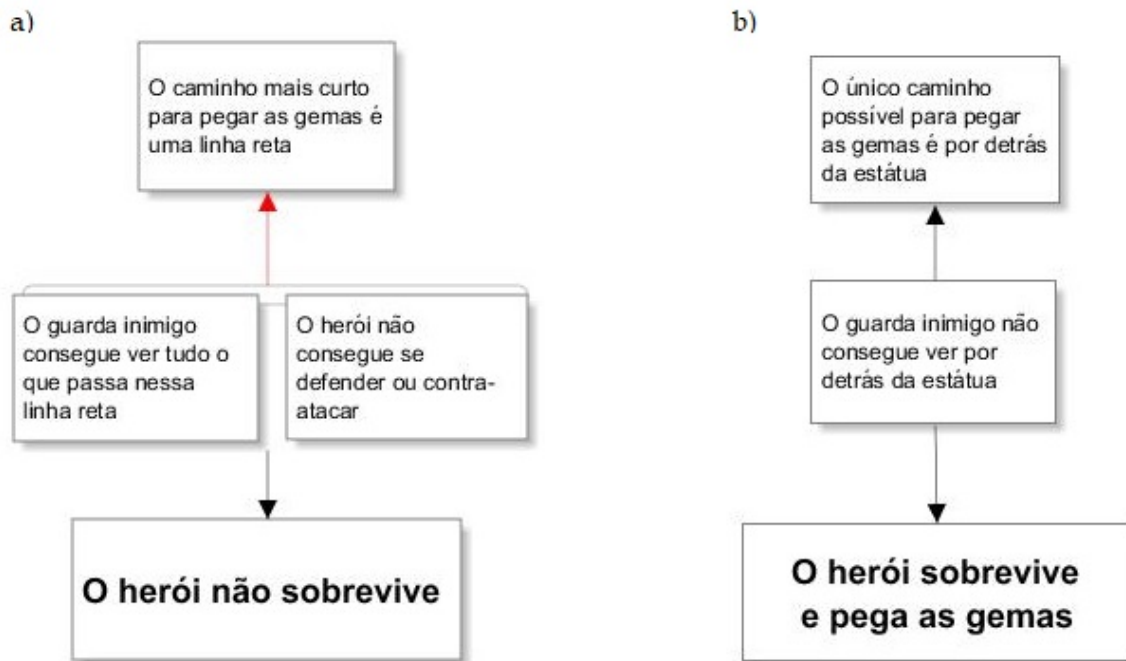
4.1.2.4 Guarda Sombrio: pensamento lógico e algorítmico em ação conjunta

A partir de Beecher (2017, p. 37), é possível descrever o pensamento lógico como uma linha de raciocínio que distingue argumentos corretos dos incorretos e que chega a uma conclusão a partir da verificação desses argumentos. Assim como os demais níveis, “Guarda Sombrio” requer que o jogador utilize seu pensamento lógico para elaborar uma linha de raciocínio que cumpra seus objetivos. Apesar disso, esse é o primeiro momento do jogo em que o jogador precisa elaborar uma linha de raciocínio um pouco mais sofisticada.

CodeCombat é claro ao informar que o herói deve evitar ser visto pelo inimigo se quiser sobreviver e dá pistas que auxiliam o jogador a construir uma solução para o problema, como se pode ver no resumo deste nível: “Como você não tem nenhuma arma ainda, não pode combater o Ogro Munchkin que protege o caminho. Ao invés de enfrentá-lo, tente passar atrás da estátua para que ele não te veja. Assim você pode alcançar a gema e correr até o final sem ser detectado” (ver Figura 8). Não obstante, como observado nas gravações das oficinas, alguns jogadores tendem a não seguir as instruções fornecidas pelo jogo, o que significa que tais avisos podem ser ignorados.

Isso posto, para resolver esse problema, o jogador pode esboçar as mais variadas linhas de raciocínio, que podem resultar em conclusões corretas ou não. Alguns exemplos de linhas de raciocínio: a) chega-se à conclusão de que o herói não pode sobreviver se

Figura 9 – Duas possíveis linhas de raciocínio para resolver o Nível 3 - “Guarda Sombrio”



Fonte: elaborado pelo autor

andar em linha reta; b) o único caminho correto é por detrás da estátua (ver Figura 9).

O nível “Guarda Sombrio” também se destaca por deixar evidente a proximidade entre os conceitos de pensamento lógico e algorítmico. Se, por um lado, o pensamento lógico é utilizado na elucidação de uma linha de raciocínio (ver Figura 9), por outro, é o pensamento algorítmico que lhe dá movimento e que a enuncia em uma sequência de instruções corretamente ordenadas, em uma linguagem que um computador entenda.

4.1.2.5 *Kounter Kithwise*: níveis complementares, padrões e generalização

O nível subsequente, “*Kounter Kithwise*”¹⁰, expande o desafio proposto em “Guarda Sombrio”, caracterizando-se como o primeiro nível complementar do jogo, isto é, uma fase que tem como objetivo fortalecer os conceitos e dinâmicas trabalhados anteriormente.

Esses níveis são opcionais, não fazendo parte do percurso principal da aventura vivenciada em *CodeCombat*. Para jogá-los, o jogador pode selecioná-los no mapa de seleção de níveis (ver Figura 10). Eventualmente, o próprio jogo pode sugerir desafios complementares ao jogador, dependendo de como este está desempenhando nos níveis obrigatórios. Para tanto, *CodeCombat* parece basear-se em fatores como o tempo médio de resolução de problemas e a quantidade de erros de cada jogador.

Os níveis complementares trabalham principalmente os conceitos de reconhecimento

¹⁰ Trocadilho para “*Counter Clockwise*”, o sentido anti-horário. O prefixo “Kith” advém do nome da cidade fictícia do jogo, *Kithgard*.

Figura 10 – Mapa dos níveis do primeiro mundo de *CodeCombat*

Fonte: elaborado pelo autor

de padrões e de generalização. O desafio proposto em “Guarda sombrio” previa que o jogador aprendesse a lidar com as dinâmicas dos inimigos, passando a considerar o comportamento deles ao elaborar suas soluções. O nível seguinte, “Kounter Kithwise”, repete o mesmo padrão de seu antecessor, mas apresenta-o de uma forma diferente, um pouco mais complicada. Assim, o jogo oportuniza que o jogador reconheça esse padrão e, mais do que isso, generalize sua resolução ao deparar-se com problemas similares. Por fim, a noção de desafios complementares como este também demonstra como o *design* de *CodeCombat* leva em consideração o ritmo de aprendizagem dos jogadores, evidenciando uma das potencialidades dos *serious games* apontada por Pivec e Pivec (2009): cada jogador progride no jogo e aprende em seu próprio ritmo.

4.1.2.6 Passos Cuidadosos: desafios de conceito como avaliação em *CodeCombat*

O nível “Passos Cuidadosos” é o primeiro desafio de conceito de *CodeCombat*, um tipo especial de nível complementar. Assim, além de trabalhar os conceitos de reconhecimento de padrões e generalização, ele aumenta consideravelmente o nível de dificuldade do desafio, como se fosse o primeiro “chefão” do jogo. Ao iniciá-lo, o jogador depara-se com dois avisos (também disponíveis no material do professor):

1. “Este é um desafio de conceito. É hora de verificar o seu conhecimento dos comandos básicos. Recolha todas as gemas e volte para a saída, que é marcada com a marca X vermelha”;

2. “Desculpe, não há dicas aqui :-(. É aqui que você usa o que aprendeu nos níveis anteriores. Se você tiver problemas com essa tarefa, volte e analise os últimos níveis para ter certeza de que os entende!”.

Consideramos que o jogo utiliza os desafios de conceito como avaliação, no sentido de “verificar” a aprendizagem: caso o jogador consiga resolver tais desafios, significa que está aprendendo; caso contrário, não está aprendendo ou está com dificuldades. Por conta disso, é admissível conjecturar que o jogo busca avaliar o nível de desenvolvimento real dos jogadores, isto é, os produtos do desenvolvimento, aquilo que eles conseguem fazer por conta própria em um dado momento (VIGOTSKI, 2007), em vez de atuar na ZDP, como se esperaria de um mediador *vigotskiano*. No entanto, contra-argumentamos que, na prática, *CodeCombat* atua como elemento mediador, como outro mais experiente em relação ao jogador, e que, portanto, leva em consideração tanto o desenvolvimento real quanto a ZDP. Numa perspectiva vigotskiana, isso é muito importante, pois aquilo que hoje é realizado somente com a assistência de outro mais experiente, amanhã será desenvolvimento real, tornando-se parte das aquisições do indivíduo, por meio da internalização (VIGOTSKI, 2007).

Percebemos, assim, que o *serious game* atua intencionalmente como elemento mediador da aprendizagem, pois ele não apresenta desafios desconexos aos jogadores na esperança de que eles consigam resolvê-los e aprender os conceitos ao acaso. Em vez disso, *CodeCombat* segue uma certa estratégia de mediação para que, sob sua perspectiva, o jogador aprenda. Por um lado, seus níveis preparatórios, que introduzem os conceitos, procuram dar acesso ao objeto de conhecimento por meio de todos os recursos disponíveis (sugestões, vídeos, pistas e dicas), possibilitando que o jogador construa, aos poucos, uma compreensão daquilo que está experimentando no jogo – para que ocorram os primeiros movimentos de internalização. Por outro, seus desafios de conceito procuram estimar o nível de desenvolvimento dos jogadores, de modo que ele perceba quando é necessário propor-lhes níveis complementares e, ao mesmo tempo, forneça indicadores ao mediador da turma no Painel do Professor.

4.1.2.7 Mina Inimiga: aprofundando a avaliação da solução de um problema

Após analisar, decompor e resolver um problema, tendo implementado e testado uma solução para ele, o jogador pode partir para o próximo desafio, correto? Não necessariamente! Um dos conceitos relacionados ao pensamento computacional é a avaliação. Conforme Beecher (2017, p. 135), antes de tomar-se um problema como resolvido, deve-se assegurar que a solução para este possui qualidade, que é uma boa solução. E isso implica em uma avaliação, em questionar-se: a solução está correta? Ela realmente resolve o problema em questão? Ela é eficiente? Ela é elegante, sendo simples e ainda eficaz?

Figura 11 – *Screenshot* do Nível 4 - “Mina Inimiga”, primeiro nível a exigir uma solução eficiente



Fonte: elaborado pelo autor

De modo geral, todos os níveis de *CodeCombat* contém pelo menos um destes aspectos, a exatidão (ou *correctness*, a qualidade de estar correto). Isso significa que sempre haverá verificação da exatidão da solução elaborada pelo jogador. O *feedback* para essa verificação é dado em forma de animação dos personagens e do cenário, que reproduz os passos do algoritmo criado; os comandos descritos pelo jogador são executados pelo herói, ao passo que os inimigos e armadilhas reagem às ações deste. Ao final da animação, o jogador descobre se os objetivos do nível foram cumpridos ou não, pois o jogo exibe uma mensagem de sucesso em caso de êxito ou apresenta razões para a solução ser considerada incorreta – complementar a isso, o próprio estado resultante dos personagens e do cenário após a execução do algoritmo é capaz de comunicar a exatidão da solução.

Não obstante, há níveis, como o “Mina Inimiga” (ver Figura 11), que vão além do aspecto da exatidão, exigindo que o jogador se preocupe com a eficiência de suas soluções. Para tanto, requer que a solução do problema seja descrita em, no máximo, cinco instruções.

Se não houvesse um limite de instruções permitidas, uma possível solução para este nível seria a que está descrita no Código 3. Essa solução estaria correta, pois o herói escaparia das minas inimigas e coletaria a gema no fim da masmorra. Entretanto, notamos que há repetição de várias instruções, como `moveRight`, que é chamada sequencialmente em duas oportunidades, e `moveDown`, chamada três vezes em sequência. No vácuo, essa ressalva pode parecer insignificante. Porém, otimizar é essencial num cenário em que os


```
1 hero.moveRight()  
2 hero.moveRight()  
3 hero.moveUp()  
4 hero.moveRight()  
5 hero.moveDown()  
6 hero.moveDown()  
7 hero.moveDown()  
8 hero.moveRight()  
9 hero.moveRight()
```

Código 3 – Solução não-otimizada para o nível “Mina inimiga”

```
1 hero.moveRight(2)  
2 hero.moveUp()  
3 hero.moveRight()  
4 hero.moveDown(3)  
5 hero.moveRight(2)
```

Código 4 – Solução ótima para o nível “Mina inimiga”

recursos computacionais são limitados, onde cada instrução “economizada” pode fazer a diferença.

Para explicitar a possibilidade de otimização das soluções, o jogo introduz neste nível o conceito de argumentos, informações adicionais passadas a uma função para modificar o que ela faz. No caso específico das funções de movimento, o argumento informa a quantidade de passos que o herói deve caminhar. Com a utilização de argumentos, é possível otimizar a solução anterior de modo que se utilizem apenas cinco instruções, em vez de nove (ver Código 4).

Limitar o número de instruções pode parecer uma decisão arbitrária por parte dos *game designers* de *CodeCombat*, mas cumpre um papel muito importante ao ensinar uma nova mecânica que pode ser bastante útil ao longo do jogo, os argumentos. Intencionalmente ou não, do ponto de vista do PC, essa mecânica parece corroborar o desenvolvimento do conceito de avaliação.

4.1.2.8 Nomes Próprios: atacando, reconhecendo padrões e generalizando

O nível “Nomes Próprios” (ver Figura 12) introduz um novo comando: **attack** (ataque). A partir desse momento, o herói não precisa mais se esconder dos inimigos; ao invés disso, pode atacá-los impiedosamente. Porém, ele precisa saber o nome do alvo de seus ataques. Para tanto, o jogador precisa informar o nome do inimigo alvo como

Figura 12 – *Screenshot* do Nível 5 - “Nomes Próprios”, em que o comando de ataque é introduzido



Fonte: elaborado pelo autor

argumento da função: `attack("Nome de inimigo")`.

Para explicar isso ao jogador, o jogo disponibiliza a seguinte dica em uma de suas sugestões:

Você deve saber algumas coisas para conseguir terminar este nível:

1. É preciso atacar cada Ogro Munchkin **duas vezes** para derrotá-lo;
2. Digite os nomes corretamente, respeitando as letras maiúsculas e minúsculas! “Brak” e “Treg”;
3. Coloque os nomes entre aspas para torná-los strings;
4. Derrote “Brak”, mova-se para a direita e pegue a gema;
5. Depois, derrote “Treg” atacando ele duas vezes;
6. Não tem problema se for derrotado, você sempre pode continuar tentando.

Talvez essa dica esteja explicando mais do que deveria, no sentido de que ela descreve por completo o algoritmo que resolve o problema. Por outro lado, como essa é a primeira vez em que o jogador enfrenta um inimigo – uma nova mecânica de jogo –, isso é compreensível. Ademais, nessa dica constam informações valiosas ao jogador, como o número de ataques necessários, bem como a forma em que os nomes dos inimigos devem ser informados (entre aspas).

Assim como as funções de movimentação (`move`), o comando de ataque recebe um argumento. Contudo, há pelo menos duas diferenças entre eles: o nome do inimigo é um argumento obrigatório e é uma *string* (cadeia de caractere), um tipo de dado que

representa texto. Desta forma, há um padrão a ser reconhecido, que é o uso de argumentos, ao mesmo tempo em que há novos fatores a considerar, como o tipo de dado.

Outro destaque acerca deste nível também envolve reconhecimento de padrões. Mais precisamente, a ausência de um possível padrão: diferentemente das funções de movimentação, que recebem como argumento o número de repetições da instrução, a função de ataque não permite informar o número de repetições, apesar de serem necessários mais do que um único ataque para derrotar os inimigos. A ausência de um argumento que informe a quantidade de ataques impossibilita a generalização dessa noção por parte do jogador, que, a essa altura do jogo, está acostumado a movimentar-se de forma mais otimizada. Como pudemos perceber ao longo da oficina, os jogadores realmente sentiram falta de tal argumento.

Por fim, ainda neste nível, cabe ressaltar que os conceitos de abstração e modelagem estão presentes, mesmo que de forma implícita, pois tanto o número de passos (número inteiro) que o herói deve dar, quanto o nome dos inimigos (*string*) são modelos, representações abstratas de uma ideia do mundo real no mundo virtual do jogo.

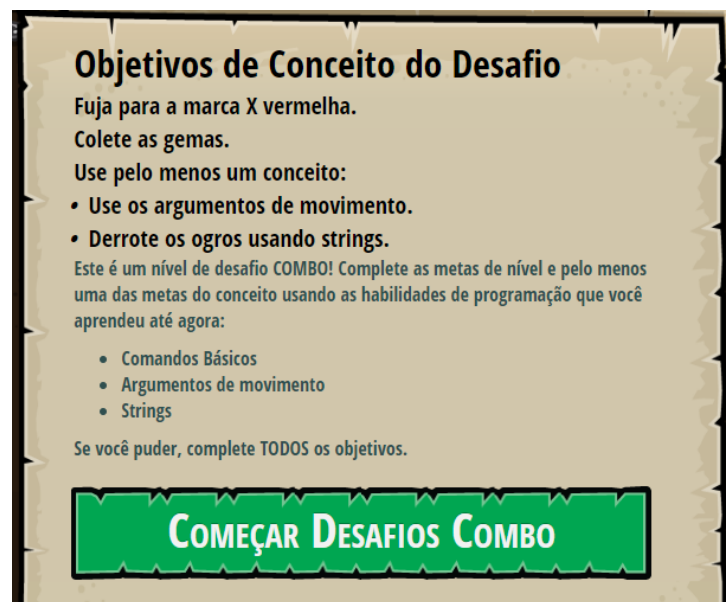
4.1.2.9 Hora de Dormir: decomposição em desafios *combo*

Tal como os desafios de conceito, os desafios *combo* são níveis especiais cujo objetivo é avaliar a aprendizagem dos conceitos apresentados até um certo momento. A diferença entre os dois é que os desafios *combo* trazem consigo mais do que um conceito por vez. O nível “Hora de Dormir”, por exemplo, abrange dois conceitos: argumentos e *strings* (ver Figura 13).

Assim como nos demais níveis, antes do jogador iniciar o desafio, *CodeCombat* exibe os objetivos de maneira destacada, à direita da tela. No caso de “Hora de Dormir”, não somente são listados os objetivos do nível, mas também se menciona a necessidade de utilização de pelo menos um dos conceitos estudados. Como não há obrigação de se utilizar os dois conceitos ao mesmo tempo, o jogo não exige que os inimigos sejam derrotados (pois estão dormindo) e tampouco limita o número de instruções utilizadas. Por outro lado, o jogo convida o jogador a completar todos os objetivos, caso ache interessante e viável.

Embora haja tal flexibilidade na formulação do problema, ainda existem pelo menos dois subproblemas a resolver: a) coletar as gemas e b) derrotar os inimigos. Por conta disso, o jogador tem a oportunidade de abordar esse desafio utilizando uma estratégia de decomposição. Caso o fizer, deverá resolver primeiro o subproblema da captura das gemas, que requer movimentar-se no cenário descrevendo uma trajetória que colete todas as quatro gemas, evitando os bloqueios no caminho (ver Figura 14). Nesta etapa de formulação do subproblema, se os comandos de movimentação forem designados com o uso de argumentos, o problema como um todo estará resolvido, pois um dos conceitos terá sido utilizado. Caso contrário, ou se desejar derrotar os inimigos, completando todos os objetivos do nível, o

Figura 13 – Objetivos do desafio combo “Hora de Dormir”: comandos básicos, argumentos e *strings*



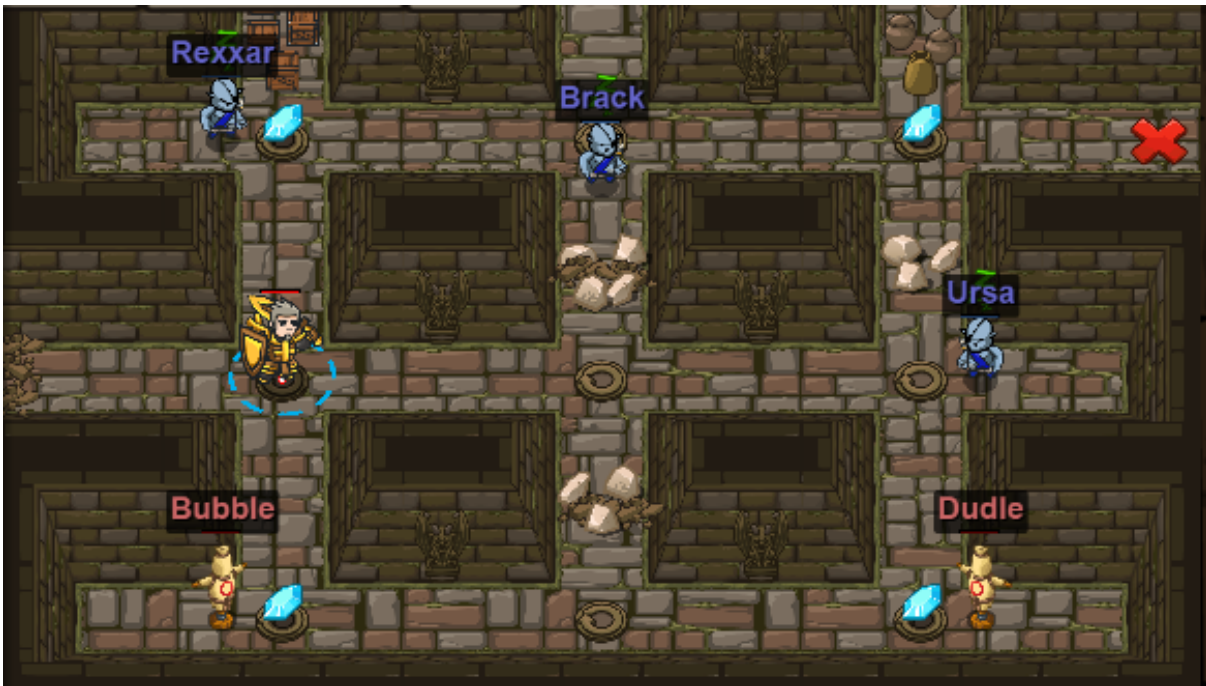
Fonte: elaborado pelo autor

```
1 hero.moveRight(2)
2 hero.attack("Ursa")
3 hero.attack("Ursa")
4 hero.moveDown()
5 hero.moveLeft(2)
6 hero.moveUp(2)
7 hero.attack("Rexxar")
8 hero.attack("Rexxar")
9 hero.attack("Brack")
10 hero.attack("Brack")
11 hero.moveRight(2)
```

Código 5 – Solução ótima para o Desafio Combo “Hora de Dormir”

jogador precisará resolver o segundo subproblema, que envolve derrotar os inimigos.

Além de convidar o jogador a utilizar decomposição de problemas, o jogo requer a elaboração de um algoritmo com instruções designadas na sequência correta para que os subproblemas sejam resolvidos simultaneamente. Uma das soluções possíveis para o desafio combo em sua completude – talvez a mais otimizada, com apenas 11 instruções – está descrita no Código 5.

Figura 14 – *Screenshot* do Desafio Combo “Hora de Dormir”

Fonte: elaborado pelo autor

4.1.2.10 O Prisioneiro: tentativa e erro, criatividade e pensamento computacional

Do mesmo modo que os demais desafios do jogo, “O Prisioneiro” (ver Figura 15) tem objetivos claramente definidos: libertar o prisioneiro, derrotar o guarda e pegar o diamante. Porém, o *design* deste nível traz elementos que não são tão óbvios num primeiro momento: o guarda que protege a *dungeon* é mais forte e resistente que aqueles encontrados pelo jogador até então, de modo que o herói não consegue derrotá-lo sozinho. Além disso, a ajuda do prisioneiro não garante uma investida bem-sucedida, pois um arqueiro como ele não tem nenhuma chance em um duelo de curta-distância. Assim, o jogador precisa elaborar uma solução que evite um confronto direto individual com o inimigo e que também lance mão da capacidade do arqueiro de atacar à distância. Ou seja, não somente deve o jogador identificar os objetivos e resolver um problema estático, como ele precisa considerar o dinamismo do ambiente e o comportamento inesperado dos inimigos ao elaborar sua estratégia, valendo-se tanto de seu pensamento computacional quanto de sua criatividade.

Percebemos, dessa forma, que tal nível aborda diretamente os conceitos de pensamento lógico, decomposição e pensamento algorítmico. No que se refere ao pensamento lógico, há diversas linhas de raciocínio possíveis para solucionar este desafio. Uma linha de raciocínio correta deve considerar que:

- sozinho, o jogador não é capaz de derrotar o Ogre inimigo;
- enquanto o aliado estiver preso, o herói não poderá vencer;

Figura 15 – “O Prisioneiro”: além de pensamento computacional, o nível exige criatividade e muitas tentativas



Fonte: elaborado pelo autor

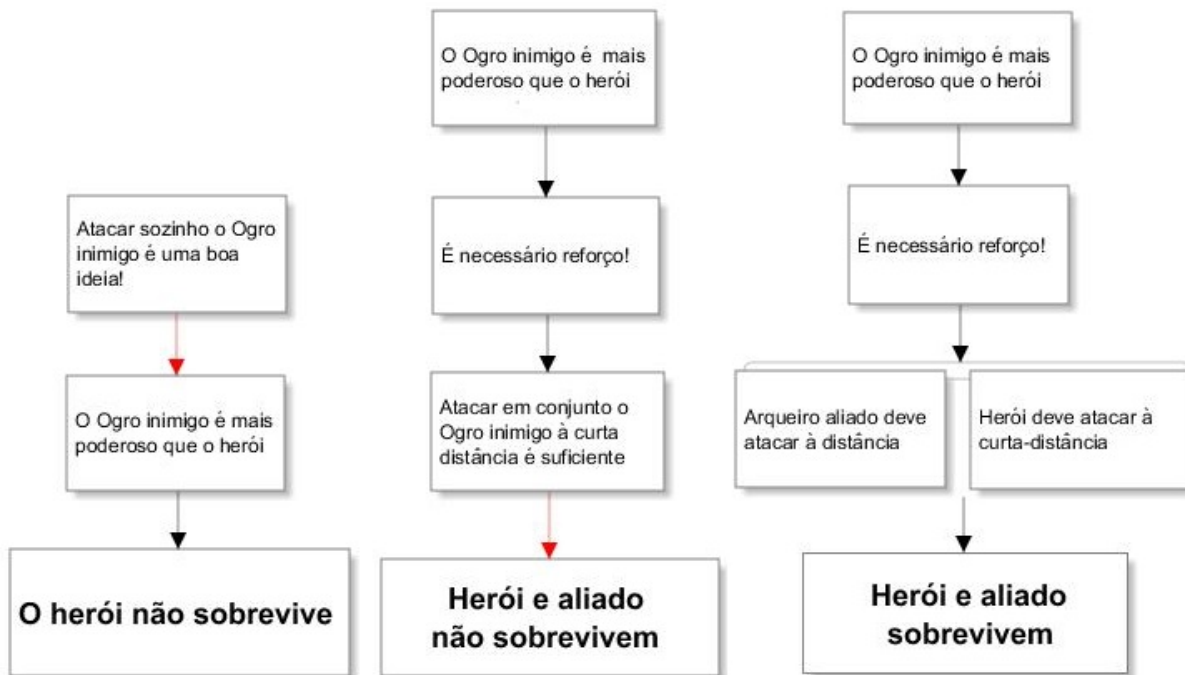
- os dois heróis devem sobreviver;
- o inimigo deve ser derrotado.

Até encontrar uma linha de raciocínio que solucione o problema, é provável que haja várias tentativas e, conseqüentemente, diversos erros por parte do jogador. Nesse aspecto, um ponto positivo do *serious game* é que ele possibilita que tais hipóteses sejam testadas instantaneamente, devido ao seu *feedback* imediato: tão logo o jogador termina de escrever seu algoritmo, é exibida uma animação dos comandos atribuídos ao herói, bem como as reações dos inimigos e do cenário. Esse CMI permite ao jogador perceber quais pontos de sua linha de raciocínio estão falhando e o que ele deve modificar para suceder.

Nesse sentido, a decomposição pode ser uma estratégia útil para que o jogador compreenda e resolva o problema por completo, servindo também como complemento para a abordagem de tentativa e erro, tornando-a metódica, ao invés de aleatória. Do mesmo modo, a sequência das ações é vital para a resolução deste problema: primeiro, o herói deve pedir ajuda de um aliado; depois, é necessário que o herói se movimente de forma sincronizada com esse aliado para que não seja derrotado pelo inimigo; por último, herói e aliado podem capturar a gema livremente.

Logo, evidenciamos como há um forte vínculo entre decomposição, pensamento lógico e pensamento algorítmico. A decomposição facilita a resolução de um problema ao dividi-lo em subproblemas, enquanto o pensamento lógico cria uma linha de raciocínio para

Figura 16 – Algumas das linhas de raciocínio possíveis para resolver o problema d'O Prisioneiro



Fonte: elaborado pelo autor

resolvê-lo. O pensamento algorítmico, por sua vez, é necessário para colocar em prática a linha de raciocínio que resolve os subproblemas.

Finalmente, destacamos que o nível “O Prisioneiro” também concerne o conceito de generalização, uma vez que oportuniza a reutilização de todos os comandos aprendidos até então: movimento (*move*), ataque (*attack*) e comunicação (*say*).

4.1.2.11 *Loop* no Armazém: reconhecendo padrões (onde parece não haver)

Um laço de repetição (*loop*) é “um procedimento ou série de instruções em um programa de computador que são executadas repetidamente até que um teste mostre que uma condição específica foi alcançada ou até que o programa seja concluído” (LOOP, 2004, p. 199, tradução do autor¹¹). *Loops* são ubíquos na programação de computadores, pois permitem que instruções sejam executadas repetidamente sem a necessidade de escrevê-las mais do que uma vez. O primeiro contato do jogador com o conceito de *loop* acontece no nível 9, “Dança no Fogo”, em que ele conhece o comando `while True` (enquanto verdade for verdade), que pode ser utilizado para repetir indefinidamente uma série de instruções. No contexto de tal desafio, o herói deve ficar eternamente fugindo de bolas de fogo lançadas pelo inimigo (ver Código 6).

¹¹ “a procedure or series of instructions in a computer program that are performed again and again until a test shows that a specific condition has been met or until the program is completed.”

```
1 while True:
2     hero.moveRight()
3     hero.moveLeft()
```

Código 6 – Solução para o nível 9 - “Dança no Fogo”

A partir dessas instruções, o herói ficará indefinidamente movimentando-se à direita e à esquerda, desviando dos projéteis lançados pelo inimigo. Esse tipo de *loop*, que não possui condição de parada, é conhecido como *loop* infinito. No primeiro mundo de *CodeCombat*, parte do *serious game* que está no escopo desta pesquisa, os *loops* são todos infinitos. Isso ocorre por escolha dos desenvolvedores, que introduzem a estrutura de controle **if-else** (se-senão), responsável pela aferição de veracidade argumentos lógicos, apenas no segundo mundo do jogo. Portanto, pode-se dizer que o jogador é introduzido explicitamente ao conceito de *loop* e, implicitamente, à noção de condições de parada, que envolvem a aferição de veracidade de argumentos (**while True** = *enquanto verdade for verdade*).

Se por um lado o conceito de *loop* é apresentado no nível “Dança no Fogo”, por outro, seu entendimento é testado pra valer no desafio de conceito “*Loop* no Armazém” (ver Figura 17). Nesse desafio, o jogador precisa encontrar a saída do armazém (uma masmorra bastante ampla) em no máximo cinco instruções. Logo fica óbvio ao jogador que não é possível resolver tal problema sem utilizar um laço de repetição. Contudo, quais instruções devem ser repetidas nesse *loop*? Essa é uma questão um tanto quanto complicada de se responder, como se comprovou em um encontro da oficina.

Nos níveis que envolvem laços de repetição, não somente se percebe a generalização do uso de *loops* infinitos, como também o reconhecimento de padrões – o conjunto exato de instruções - que resolvem os problemas. Possivelmente, o padrão encontrado no nível “*Loop* no Armazém” seja o mais complicado do primeiro mundo de *CodeCombat* (ver Figura 18 e Figura 19).

Esse desafio de conceito é mais um exemplo de nível que exige uma sequência de passos perfeita para que o problema seja resolvido (Código 7). Isto posto, “*Loop* no Armazém” pode ser considerado um nível desafiador (devido à dificuldade aparente por parte de todos os alunos) que concilia pensamento lógico, pensamento algorítmico, decomposição, generalização, reconhecimento de padrões e avaliação – quase todos os principais conceitos relacionados ao PC.

Figura 17 – Screenshot do desafio de conceito “Loop no Armazém”



Fonte: elaborado pelo autor

Figura 18 – Reconhecimento de padrões: série de instruções que, quando repetidas, solucionam o “Loop no Armazém”



Fonte: elaborado pelo autor

Figura 19 – Momento inicial da animação da trajetória do herói no armazém após a elaboração de uma solução correta



Fonte: elaborado pelo autor

```
1 while True:  
2     hero.moveUp(2)  
3     hero.moveRight(2)  
4     hero.moveDown()  
5     hero.moveLeft()
```

Código 7 – Solução para o desafio de conceito “Loop no Armazém”

4.1.3 Mediação

Esta categoria está relacionada ao papel mediador do professor como alguém mais experiente, responsável por organizar a experiência de aprendizagem – em um contexto mediatizado por um *serious game*. Retomando os norteadores teóricos, lembramos que a acepção de aprendizagem utilizada nesta pesquisa, *obuchenie*, é um amálgama dos processos de ensinar (organização do ambiente) e aprender (esforço individual que resulta em mudança nos processos psicológicos e no conhecimento do indivíduo) (COLE, 2009). É importante trazer à tona tal conceito, pois como esta pesquisa investiga como *serious games* podem desenvolver o pensamento computacional, pode parecer contraproducente argumentar que eles por si mesmos não sejam suficientes para tal finalidade. A noção de *obuchenie* admite que a organização do ambiente influencia diretamente na experiência de aprendizagem do sujeito, seja esta mediatizada por um *serious game* ou não. Logo,

a questão da mediação do professor – um MKO – também faz parte do escopo desta pesquisa.

4.1.3.1 Mediação, MKO, ZDP e internalização

“Ensinar uma criança o que ela não é capaz de aprender é tão estéril quanto ensiná-la a fazer o que ela já faz sozinha” (VIGOTSKI, 2009, p. 336-337). É possível que um jogo apresente situações que fogem completamente do nível de desenvolvimento do aluno. Ao mesmo tempo, o mesmo jogo pode acabar ensinando algo que o aluno já sabe, o que tampouco é frutífero em termos de aprendizagem. É nesses casos em que a mediação do professor (ou MKO) se revela necessária.

Quando o mediador percebe que a situação de jogo excede o nível de desenvolvimento real do sujeito, ele pode intervir e preencher as lacunas existentes para que, em seguida, o problema esteja ao alcance de sua ZDP. No excerto abaixo, percebemos a tentativa do professor em explicar a **Ermac** o conceito de variável, um dos mais abstratos do jogo:

PROFESSOR : Eu ia passar isso no vídeo (*com tradução simultânea, no início do encontro*), mas vi que vocês tavam meio aborrecidos já! (*risos*)

PROFESSOR : Não sei se vocês já viram isso na escola já, mas tipo, VARIÁVEL é qualquer coisa que muda de valor. Tipo, não sei se tu já chegou a ver álgebra na escola, tipo, “X” e essas coisas, na aula de matemática?

ERMAC : (*confirma com a cabeça*)

PROFESSOR : X é uma variável, aonde X pode ser qualquer valor. Na “ciência” isso também pode... Num experimento científico, por exemplo, tu quer saber quanto é preciso de água pra regar uma planta, por exemplo. Aí tipo, essa quantidade de água é VARIÁVEL. Não se sabe o valor ao certo. Pode ser 2 litros de água, ou 3 litros de água ou 4 litros de água. Então, tu tem lá uma variável que é assim: *quantidadeDeAgua* ou poderia ser tipo *agua*, por exemplo (*demonstra isso escrevendo no editor do jogo: $agua = 2$, depois $agua = 3$, $agua = 4$*). Até aí, beleza?

ERMAC : (*confirma com a cabeça*)

PROFESSOR : Aqui a gente não tá lidando com experimento científico... mas é um problema que também utiliza variável. Esse cara aqui (*atribuição na tela: $enemy1 = \text{“Kratt”}$*), ele tá dizendo *enemy1*, tá dizendo assim: nosso inimigo, o primeiro – poderia ser qualquer nome, poderia ser *inimigo*, em português – aí o nosso inimigo tem o nome de “Kratt”. Então, em vez de colocar o nome do Kratt, tu vai colocar *enemy1*. Tu tá colocando (*o nome*) na variável do “inimigo 1” e o valor dela é “Kratt”. Assim como eu tava dizendo antes que na *água* poderia ser 2, 3 e 4.

PROFESSOR : Tu usa (*uma variável*) pra salvar um valor. Nesse caso aí, é o nome do teu

inimigo. Nesse caso aí, tu derrotou um inimigo né?! O Kratt. Nessa fase aqui, é mais pra (*o jogo*) introduzir o conceito. Mas vai ter uma fase, depois dessa, que tu não sabe o nome do inimigo. Então tu não tem como vir aqui (*no editor*) e colocar “ataque o Kratt” (*hero.attack(“Kratt”*)); tu não sabe qual é que é o nome do cara. Então, pra fazer isso, tu vai ter um poder que descobre o nome do cara. Aí tu tem que salvar isso num lugar, daí tu vai salvar isso numa variável.

Notamos que o jogo falhou em explicar o conceito de variável, pois assumira que o aluno havia visto e compreendido o vídeo que o explicava. Não somente estava o material na língua inglesa, sem legendas para qualquer outro idioma, como ainda existia a possibilidade de que o jogador não o compreendesse. Para preencher tal lacuna, o professor teve que atuar na ZDP de Ermac, encontrando nele um conhecimento prévio, um saber em comum que lhe permitisse alcançar o nível de conhecimento necessário para resolver o problema – nesse caso em específico, foi a noção de variável estudada nas aulas de álgebra do Ensino Fundamental que permitiu a Ermac seguir em frente.

Por outro lado, quando o mediador percebe que a situação de jogo é muito fácil para o aluno, ele pode desafiá-lo a abordar o problema de outras formas, propor novas atividades, ou até mesmo solicitar que ele auxilie seus colegas, como nas situações narradas anteriormente, envolvendo os sujeitos **Knight** e **Donalds**.

Outra questão pertinente à mediação observada ao longo da oficina é a atribuição de sentido, a significação – assim como a própria internalização: vários desafios do jogo foram resolvidos sem que os jogadores os compreendessem em sua plenitude, isto é, sem que atribuíssem sentido àquilo que estavam fazendo. Desta forma, percebemos uma das limitações do *serious game CodeCombat*: se por um lado o CMI e a sociointeração potencializam a resolução dos desafios do jogo, por outro, permitem que os problemas sejam resolvidos mesmo quando o jogador não os compreende inteiramente. Se um problema não for compreendido, é improvável que haja internalização do princípio que está por trás de seu propósito (os princípios do pensamento computacional, neste caso). Essa também é uma das lacunas que torna necessária a mediação do MKO.

A partir de Vigotski (2007; 2009), podemos sugerir que, para possibilitar compreensão, significação e internalização mais completas, o *serious game* também precisaria atuar na ZDP do jogador. Como visto há pouco, o jogo *CodeCombat* é capaz de adaptar os caminhos que cada jogador deve percorrer ao longo da aventura, oferecendo *feedback* dinâmico quando percebe alguma dificuldade do aluno ou erro em sua proposta de solução. O *serious game* também propõe níveis complementares aos jogadores quando julga que eles precisam aprofundar um determinado conceito. Porém, a dinamicidade desse *feedback* é limitada, uma vez que o jogo não consegue dialogar livremente com o aluno e, portanto, não consegue adequar-se à sua ZDP. O MKO, por sua vez, consegue fazê-lo. Assim, é plausível

defender que a mediação do MKO seja importante para complementar a mediatização do *serious game*, pois ela fortalece a compreensão e a atribuição de sentido do aluno em relação aos desafios do jogo, possibilitando que ele internalize os conceitos empregados em suas soluções.

4.1.3.2 A abordagem *vigotskiana* do mediador

Na maioria das vezes, os jogadores conseguiram deduzir as regras do jogo por conta própria – até mesmo os comandos em língua inglesa –, em vez de aguardarem por instruções do professor. Não obstante, para alguns jogadores, como **Ichigo**, a compreensão das regras ocorreu de forma mais lenta, de tal modo que o professor precisou auxiliá-los a compreender o sistema de jogo em mais de uma oportunidade, como nesta em que trabalhava em dupla com **Colega**:

PROFESSOR : (*observa por um tempo*) Ah tá. Qual que é o plano de vocês, assim, sem usar os códigos?

ICHIGO : A gente tá pensando em fazer duas coisas: ...

PROFESSOR : Como assim?

ICHIGO : (*Explica, com certa dificuldade, o problema que enfrentavam*)

PROFESSOR : Por exemplo: se tu escrever duas vezes para baixo, ele [herói] não vai parar. Tipo, se tu falar assim, tu tem que passar todo o TRAJETO para ele [herói] (*tenta explicar*)

ICHIGO : A gente tem que escrever duas vezes, é isso?

PROFESSOR : Não, não. Não precisa escrever duas. Tipo, [dificuldade em procurar uma forma de melhor explicar], assim ó: vamos pensar em qual caminho que vocês fariam pra ...

DUPLA : ...

PROFESSOR : (*gestualmente*) Vocês viriam aqui, aqui e ...

DUPLA : Aqui?!

PROFESSOR : Exato. E tipo, olhando assim agora, vocês devem ir pra baixo, quantas vezes?

ICHIGO : Duas.

PROFESSOR : Duas. Depois que vocês vão pra baixo duas vezes, vocês tem que ir ...

COLEGA : Uma pra cima ... e outra pra direita.

PROFESSOR : (*confirma*) Então vocês tem que escrever esses 5 comandos aí: pra baixo, pra baixo, direita, cima, direita.

ICHIGO : Hummm... entendi.

PROFESSOR : Só depois que ele [herói] vai executar.

ICHIGO : Tem que dar todos os comandos.

PROFESSOR : Sim. Aquele espaço [editor] tu usa pra escrever todo o teu plano, depois tu roda. Ele não é... Eu entendi o que tu queria dizer [sobre rodar um passo por vez]. Agora acho que vai ficar mais fácil.

Nesse excerto, percebemos o esforço do professor para compreender a dúvida dos alunos. A dupla estava tentando dar comandos *enquanto* a execução do programa acontecia, como se precisassem codificar rapidamente antes que o herói realizasse alguma ação por conta própria. Claramente a dupla ainda não havia entendido como o jogo funcionava, pois não haviam compreendido que o herói somente executava aquilo que havia sido programado de antemão. Esse é um tipo de barreira muito difícil para o *serious game*, pois nenhum de seus recursos foi suficiente para a compreensão das regras por parte desses sujeitos, tanto que o professor precisou intervir. Novamente, destacamos a importância do mediador em situações como essa.

Ademais, as principais dúvidas dos alunos eram sobre como comunicar ao jogo o plano que haviam traçado mentalmente. Nesses casos, o mediador costumava sugerir que eles expressassem suas ideias em palavras ou até mesmo apontando para a tela, em vez de usar a linguagem de programação. Dessa forma, era possível ajudá-los sem resolver os problemas em seu lugar – uma abordagem coerente com a proposta *vigotskiana*:¹²

transmitir deliberadamente novos conceitos e novas formas ao aluno é, estou convencido, tão impossível e inútil quanto ensinar uma criança a andar segundo as leis do equilíbrio. Qualquer tentativa dessa espécie apenas desvia o aluno do objetivo proposto, como a força bruta do homem que, tentando ajudar uma flor a desabrochar, passasse a desenrolá-la pelas pétalas e amassasse tudo ao redor (TOLSTOI apud VIGOTSKI, 2009, p. 146)

Finalmente, percebemos como a mediação complementa a mediatização do *serious game*: embora o jogo *CodeCombat* possua características que possibilitem o desenvolvimento do pensamento computacional de forma independente, a organização do ambiente por parte do professor-mediador ainda é importante – senão fundamental – para que se aproveite seu potencial ao máximo.

4.2 Análise das produções dos alunos

Conforme apresentado no capítulo do método, além da Análise Textual Discursiva, foi realizada uma análise das produções dos alunos no contexto do jogo, a partir dos

¹² Particularmente, o professor evitou ao máximo ser prescritivo quanto ao modo de jogar-se o jogo.

registros disponíveis no Painel do Professor, com o intuito de enriquecer uma possível resposta para a questão de pesquisa.

Dentre os indicadores analisados estão os programas escritos pelos alunos em cada nível, o tempo levado para resolver cada desafio, quais desafios de conceitos e combo foram resolvidos e como foram resolvidos (total ou parcialmente), bem como quais *conceitos de programação* foram aprendidos pelo aluno. Ao longo desta seção, serão discutidas as informações geradas com base nesses indicadores, de modo a esclarecer como elas contribuem para a *avaliação* da aprendizagem e do desenvolvimento do pensamento computacional dos alunos.

Finalmente, serão apresentadas informações obtidas a partir do cruzamento dos dados do Painel do Professor e da análise dos níveis do jogo, no sentido de corroborar a percepção de que o *serious game* contribuiu para a internalização de conceitos relacionados ao pensamento computacional.

4.2.1 Aprendizagem sob a ótica do jogo *CodeCombat*

Uma das principais informações do Painel do Professor de *CodeCombat* é a relação de quais níveis foram concluídos pelos alunos. Com base nela, é possível sabermos o quanto os alunos progrediram no jogo, bem como ter uma noção do ritmo de aprendizagem da turma como um todo. Ademais, ela também pode servir como substrato para uma avaliação mais profunda da aprendizagem.

Quadro 5 – Alunos x Níveis Concluídos

Nível	Título	Er	Co e Ic	Jo e Ho	XV e Ba	Kn	Do	Ch
1	Dungeons of Kithgard	X	X	X	X	X	X	X
2	Gems in the Deep	X	X	X	X	X	X	X
3	Shadow Guard	X	X	X	X	X	X	X
3a	Kounter Kithwise		X			X	X	
3b	Crawlways of Kith.		X			X	X	
Conceito	Careful Steps	X		X	X	X	X	X
4	Enemy Mine	X	X	X	X	X	X	X
4a	Illusory Interruption				X	X	X	
4b	Forgetful Gemsmith					X	X	
Conceito	Long Steps	X		X	X	X	X	
5	True Names	X	X	X	X	X	X	X
5a	Favorable Odds	X	X	X	X	X	X	X
5b	The Raised Sword		X			X	X	

(continua)

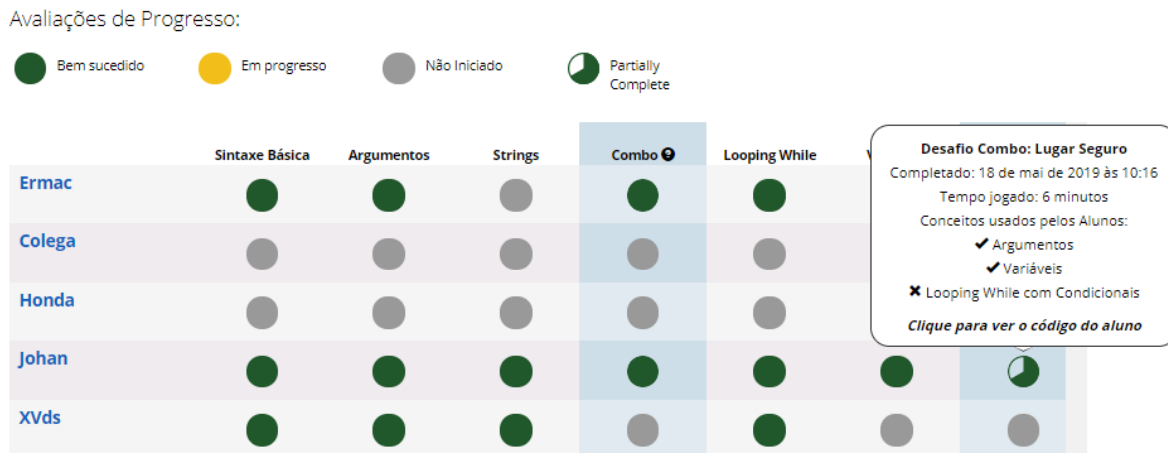
(continuação)

Nível	Título	Er	Co e Ic	Jo e Ho	XV e Ba	Kn	Do	Ch
Conceito	Dangerous Steps			X	X	X	X	X
Combo	Sleep Hour	X		X		X	X	
6	Cell Commentary	X	X	X	X	X	X	X
7	Kithgard Librarian	X	X	X	X	X	X	X
8	The Prisoner	X	X	X	X	X	X	X
9	Fire Dancing	X	X	X	X	X	X	X
10	Haunted Kithmaze	X	X	X	X	X	X	X
11	Descending Further	X	X	X	X	X	X	X
11a	Riddling Kithmaze	X	X	X	X	X	X	X
11b	Radiant Aura		X	X	X	X	X	
Conceito	Loop Warehouse	X		X	X	X	X	
12	Dread Door	X	X	X	X	X	X	X
13	Hack and Dash	X	X	X	X	X	X	
14	Cupboards of Kith.	X	X	X	X	X	X	
14a	Cupboards of Kith. A		X	X	X	X	X	
14b	Cupboards of Kith. B		X		X	X	X	
15	Known Enemy	X	X	X	X	X	X	
16	Master of Names	X	X	X	X	X	X	
16a	Lowly Kithmen	X	X	X	X	X	X	
16b	Closing the Distance	X	X		X		X	
Conceito	Master of Debug			X		X	X	
17	The Final Kithmaze	X	X	X	X	X	X	
Combo	Safe Place	X		X		X	X	
18	Kithgard Gates	X	X	X	X	X	X	
Arena	Wakka Maul	X	X	X	X	X		

Fonte: elaborado pelo autor

Na relação dos níveis concluídos pelos alunos durante a oficina, por exemplo, verificamos que a maioria dos alunos concluiu o *serious game*, com a exceção de **Cheetos** e **Donalds** (ver Quadro 5). Mas isso significa que eles não aprenderam tanto quanto os demais? No caso de Cheetos, parece razoável afirmar isso, pois ele participou apenas do segundo encontro da oficina – e isso se reflete no quadro, onde se vê que o último nível concluído por ele foi o 12º. Por outro lado, isso não se pode dizer a respeito de Donalds, uma vez que ele concluiu praticamente todos os níveis, inclusive os desafios de conceito e combos, com exceção do último.

Figura 20 – Avaliação dos desafios de conceito e combo – quais conceitos foram usados?



Fonte: *screenshot* do Painel do Professor

O próprio jogo parece levar isso em consideração, visto que ele compreende como “avaliativos” apenas os desafios de conceito e os combos. Por conta disso, *CodeCombat* oferece um indicador específico para níveis desse tipo, exibindo não somente se eles foram concluídos, mas também quais conceitos foram empregados em sua resolução.

No exemplo ilustrado na Figura 20, constatamos que **Johan** utilizou argumentos e variáveis para resolver o desafio combo “Lugar Seguro”, mas não usou *loops*. Esse é um tipo de dado cujo nível de detalhamento permite ao professor reorganizar sua estratégia. Além desses indicadores, o jogo fornece um resumo do histórico dos alunos, que lista quantos níveis foram concluídos e em quanto tempo, assim como quais *conceitos de programação* foram aprendidos (ver Quadro 6).

Destacamos que a avaliação do jogo faz menção a conceitos de programação, como argumentos, *strings* e variáveis, mas não ao pensamento computacional *per se*. Isso não implica em um problema, pois o jogo tem sua própria organização e seus próprios objetivos de aprendizagem – e não se deve esperar que os autores de um jogo se baseiem no mesmo referencial teórico e tenham a mesma visão epistemológica dos professores que utilizam tal jogo como elemento mediador. No entanto, pouco podemos dizer a respeito da aprendizagem e do desenvolvimento do pensamento computacional utilizando somente esse indicador.

Com base no Quadro 6, é possível observar que a maioria dos sujeitos da pesquisa demonstraram indícios de internalização de conceitos de programação, mas não se pode ir muito além disso – ou será que **Colega** e **Ichigo** realmente não aprenderam *nada* durante a oficina? Parece-nos evidente que é preciso ir mais fundo para asseverar-se qualquer interpretação acerca do pensamento computacional.

Quadro 6 – Conceitos aprendidos pelos alunos na visão do jogo

Aluno	Níveis	Tempo	Conceitos (com base nos desafios de conceito e combos completados)
Ermac	21	2 horas	Sintaxe Básica, Argumentos, Strings, Looping While
Colega/Ichigo	20	3 horas	
Johan/Honda	20	2 horas	Sintaxe Básica, Argumentos, Strings, Looping While, Variáveis
XVds/Bags	19	2 horas	Sintaxe Básica, Argumentos, Strings, Looping While
Knight	20	2 horas	Sintaxe Básica, Argumentos, Strings, Looping While, Variáveis
Donalds	18	uma hora	Sintaxe Básica, Argumentos, Strings, Looping While, Variáveis
Cheetos	12	34 minutos	Sintaxe Básica, Strings

Fonte: elaborado pelo autor

Assim, percebemos que os dados fornecidos pelo jogo são de grande valia para o acompanhamento e avaliação do professor, mas que eles precisam ser adaptados e interpretados de acordo com sua lente teórica – pelo menos no caso desta pesquisa, onde não parece coerente simplesmente tomá-los como verdade sobre a aprendizagem dos alunos, tanto por conta do objeto de conhecimento quanto pela abordagem *vigotskiana*.

4.2.2 Aprofundando a análise

Como visto há pouco, apesar de essenciais, os indicadores do Painel do Professor precisavam ser aprofundados para que pudessemos perceber a aprendizagem e o desenvolvimento do pensamento computacional à luz do aporte teórico desta pesquisa. Para tanto, primeiramente foi necessário mapear quais níveis abordavam quais conceitos do pensamento computacional de forma mais acentuada¹³. O resultado desse mapeamento está disponível no Quadro 7.

¹³ Tal empreendimento foi feito com base na experiência de jogo e na análise de seus recursos. Comentários sobre tais escolhas também foram tecidos durante a discussão da categoria emergente “desafio”, na seção anterior.

Quadro 7 – Níveis x Conceitos do Pensamento Computacional

Nível	Título	Lóg	Algo	Decomp	Pad. Gen	Ab. Mod	Aval
1	Dungeons of Kith.					X	
2	Gems in the Deep		X		X		
3	Shadow Guard	X	X				
3a	Kounter Kithwise				X		
3b	Crawlways of Kith.				X		
Conceito	Careful Steps				X		
4	Enemy Mine						X
4a	Illusory Interruption				X		
4b	Forgetful Gemsmith				X		
Conceito	Long Steps		X		X		X
5	True Names	X			X	X	
5a	Favorable Odds				X		
5b	The Raised Sword				X		
Conceito	Dangerous Steps				X		
Combo	Sleep Hour		X	X	X		X
6	Cell Commentary						
7	Kithgard Librarian				X		
8	The Prisoner	X	X	X	X		
9	Fire Dancing	X					X
10	Haunted Kithmaze				X		X
11	Descending Further				X		
11a	Riddling Kithmaze				X		
11b	Radiant Aura				X		
Conceito	Loop Warehouse		X	X	X		X
12	Dread Door				X		
13	Hack and Dash			X	X		
14	Cupboards of Kith.					X	
14a	Cupboards of Kith. A				X		
14b	Cupboards of Kith. B				X		
15	Known Enemy					X	
16	Master of Names					X	
16a	Lowly Kithmen				X		
16b	Closing the Distance				X		
Conceito	Master of Debug				X		X
17	The Final Kithmaze	X			X		

(continua)

(continuação)

Nível	Título	Lóg	Algo	Decomp	Pad. Gen	Ab. Mod	Aval
Combo	Safe Place				X	X	
18	Kithgard Gates		X	X			
Arena	Wakka Maul	X	X	X	X	X	X

Fonte: elaborado pelo autor

A partir disso, foi possível cruzar a relação de níveis concluídos pelos alunos (ver Quadro 5) com o mapeamento dos conceitos presentes em cada nível para saber em que medida os sujeitos aprenderam cada conceito. Para esse fim, foi elaborado um programa na linguagem *Python* (ver Apêndice E). O cruzamento dos dados por meio desse algoritmo resultou no Quadro 8.

Quadro 8 – Conceitos do Pensamento Computacional x Alunos

Conceito	Marcas	Er	Co Ic	Jo Ho	XV Ba	Kn	Do	Ch
P. Lógico.	6	100.0	100.0	100.0	100.0	100.0	83.33	66.67
P. Algo.	8	100.0	62.5	100.0	87.5	100.0	87.5	37.5
Decomp.	6	100.0	66.67	100.0	83.33	100.0	83.33	33.33
Padr. Gen.	29	65.52	68.97	75.86	75.86	96.55	96.55	41.38
Abstr. Model.	7	100.0	85.71	100.0	85.71	100.0	85.71	28.57
Avaliação	8	87.5	50.0	100.0	75.0	100.0	87.5	37.5
Total	64	92.17%	72.31%	95.98%	84.57%	99.42%	87.32%	40.82%

Fonte: elaborado pelo autor

Muito mais coerente com esta pesquisa do que o Quadro 6, que retratava os conceitos aprendidos pelos alunos na visão do jogo, esse quadro permite verificar o quanto cada sujeito aprendeu sobre cada conceito do pensamento computacional. Sua interpretação é simples: a coluna *Marcas* indica a quantidade de níveis em que um conceito apareceu de forma destacada ao longo do jogo, ao passo em que as colunas subsequentes denotam o “aproveitamento” de todos os alunos (ou duplas) naquele conceito específico (ex: **XVds** e **Bags** obtiveram o aproveitamento de 75% em *avaliação*, ou seja, completaram 6 dos 8 níveis que abordaram tal conceito). Na última linha, listamos o aproveitamento geral dos alunos no *serious game*.

Baseado nessa informação, podemos sugerir que há indícios de internalização de conceitos relacionados ao pensamento computacional e que esse processo foi mediatizado pelo *serious game CodeCombat*. Esses indícios são importantes, pois corroboram aquilo que se apresentou na Análise Textual Discursiva e enriquecem uma possível resposta à questão de pesquisa. No entanto, algumas ressalvas ainda precisam ser feitas:

- a percepção dos indícios de internalização ocorre em um nível quantitativo, binário (ou acertou, ou errou), que parece refletir o fenômeno observado com pouca verossimilhança. Sabemos que alguns alunos conseguiram resolver todos os problemas do jogo, mas isso não necessariamente comprova algo – assim como um aluno que resolve um exercício de álgebra com papel e lápis, que pode ou não ter aprendido, apesar de ter acertado a questão;
- o cruzamento dos dados atribuiu o mesmo peso a todos os níveis, mesmo sabendo da maior relevância dos desafios de conceito e combo.

Nesse sentido, para resolver essas duas questões e tornar a análise mais coerente com a visão epistemológica desta pesquisa, foi realizado um novo esforço analítico. Desta vez, foram atribuídos pesos distintos para cada nível, de acordo com sua relevância no contexto do jogo. Assim, foi possível modelar a noção de maior relevância dos desafios de conceito e combo em relação aos demais níveis. Além disso, em vez de simplesmente considerarmos uma solução como certa ou errada, preto ou branco, atentamos para os tons de cinza que traduzem melhor o sentido da produção do aluno: uma solução incompleta, mas que parece promissora vale mais do que nem tentar; ao mesmo tempo, uma solução correta que utiliza um caminho alternativo vai além do esperado, portanto merece uma avaliação melhor. Deste modo, todas as soluções, de todos os alunos, para todos os níveis, foram avaliadas atentamente e classificadas de acordo com o Quadro 9. Por fim, também foi desenvolvido um segundo programa, também na linguagem *Python*, para levar em conta a nova estrutura de dados (ver Apêndice H). Os dados de entrada desse algoritmo estão descritos no Quadro 10.

Quadro 9 – Classificações das soluções

Classificação	Nota	Descrição
OK +	1.1	Solução correta, melhor otimizada
OK Cr	1.1	Solução correta e criativa
OK *	1.05	Solução correta, caminho alternativo
OK	1	Solução correta
NO *	0.95	Solução correta, não otimizada, caminho alternativo
NO	0.9	Solução correta, mas não otimizada

MKO	0.8	Solução correta, feita em conjunto com um MKO
OK -	0.75	Solução correta, mas incompleta
NC	0.25	não concluído
-	0	não realizado

Fonte: elaborado pelo autor

Quadro 10 – Alunos x Níveis Concluídos (com classificação e ponderação)

Nível	Peso	Er	Co Ic	Jo Ho	XV Ba	Kn	Do	Ch
1	1	OK	OK	OK	OK	OK	OK	OK
2	1	NO	OK	OK	OK	NO	NO	OK
3	1	OK	OK	OK	OK	OK	OK	OK
3a	0.5	-	OK	-	-	OK	OK	-
3b	0.5	-	NO	-	-	OK	OK	-
Conceito A	2	OK*	-	OK*	OK	OK*	OK*	MKO
4	1	OK	OK	OK	OK	OK	OK	OK
4a	0.5	-	-	-	OK	OK	OK	-
4b	0.5	-	-	-	-	OK	OK	-
Conceito B	2	OK	-	OK	OK	OK	OK	-
5	1	OK	OK	OK	OK	OK	OK	OK
5a	0.5	OK	OK+	OK	OK+	OK+	OK+	OK
5b	0.5	-	OK	-	-	OK	OK	-
Conceito C	2	-	-	NO	OK	NO	NO	OK
Combo A	3	OK*	-	OK*	-	OK	OK+	-
6	1	OK	OK+	OK	OK Cr	OK+	OK+	OK+
7	1	OK	OK	OK	OK Cr	OK	OK+	OK+
8	1	NO*	OK	OK+	NO*	OK	OK	MKO
9	1	OK	OK	OK	OK	OK	NO	OK
10	1	OK	OK+	OK	OK+	OK+	NO	OK+
11	1	OK	OK	OK	OK	OK	OK	OK
11a	0.5	OK	OK	OK	OK	OK	OK	OK
11b	0.5	-	OK	OK	OK	NO Cr	OK	-
Conceito D	2	MKO	-	MKO	OK	OK	OK	-
12	1	OK	OK	OK	OK	OK	OK	OK
13	1	MKO	MKO	MKO	OK	MKO	OK	NC

(continua)

(continuação)

Nível	Peso	Er	Co Ic	Jo Ho	XV Ba	Kn	Do	Ch
14	1	OK	NO	OK	OK	OK	OK	-
14a	0.5	-	OK	OK	OK	OK	OK	-
14b	0.5	-	OK	-	OK	OK	OK	-
15	1	OK	OK*	OK	OK	OK	OK	-
16	1	OK+	OK+	OK	OK*	OK*	OK	-
16a	0.5	OK*	OK*	OK*	OK*	OK*	OK Cr	-
16b	0.5	NO	NO	-	NO	-	OK	-
Conceito E	2	-	-	OK	-	OK*	OK	-
17	1	OK	OK	OK	MKO	OK Cr	OK	-
Combo B	3	NC	-	OK-	-	OK+	NC	-
18	1	OK	OK*	OK Cr	OK Cr	OK	OK	-
Arena	3	OK	OK	OK	OK	OK	OK-	-

Fonte: elaborado pelo autor

Como resultado, obtivemos uma avaliação mais categórica, qualitativa, das produções dos alunos no jogo (ver Quadro 11). Em vez de simplesmente saber que um aluno resolveu um determinado desafio, foi possível saber *como* ele o fez, bem como avaliá-lo de uma forma mais justa. A análise minuciosa de cada solução dos alunos permitiu-nos ver nuances de criatividade, de decomposição e resolução de problemas, de reconhecimento de padrões, e de outros aspectos relacionados ao pensamento computacional – nuances esses que não seriam percebidos caso se utilizasse apenas as informações do Painel do Professor em seu estado bruto.

Quadro 11 – Conceitos do Pensamento Computacional x Alunos (com classificação e ponderação)

Conceito	Marcas	Er	Co Ic	Jo Ho	XV Ba	Kn	Do	Ch
P. Lógico.	8	99.38	100.0	100.0	96.88	100.0	89.38	47.5
P. Algo.	14	97.14	50.36	99.64	78.93	99.29	96.07	20.0
Decomp.	11	95.45	53.18	97.73	73.18	98.18	95.91	9.55
Padr. Gen.	34	68.31	49.63	86.25	70.51	97.87	91.76	34.85
Abstr. Model.	11	80.45	73.18	93.18	73.18	100.0	72.73	18.18
Avaliação	15	85.0	40.67	98.33	67.33	100.0	95.67	20.67
Total	93	87.62%	61.17%	95.86%	76.67%	99.22%	90.25%	25.12%

Fonte: elaborado pelo autor

A partir dessa segunda iteração da análise, que apresenta maior verossimilhança em relação às iterações anteriores¹⁴, temos maior propriedade para afirmar: **há indícios de internalização de conceitos do pensamento computacional, tanto em termos quantitativos quanto qualitativos, e essa internalização é mediatizada pelo *serious game CodeCombat*.**

Finalmente, concluímos esta análise salientando que, além de iniciar os processos de internalização dos jogadores, *CodeCombat* fornece ao professor ferramentas de suporte à avaliação dos alunos e isso também contribui para a aprendizagem e o desenvolvimento do pensamento computacional, mesmo que indiretamente. Os indicadores disponíveis no Painel do Professor não necessariamente se traduzem para os conceitos do pensamento computacional, mesmo assim são importantes para que se tenha registro daquilo que foi feito pelos alunos no jogo, o que possibilita análises *a posteriori*, a partir de quaisquer lentes teóricas.

4.3 Ligando os pontos: uma possível resposta à questão de pesquisa

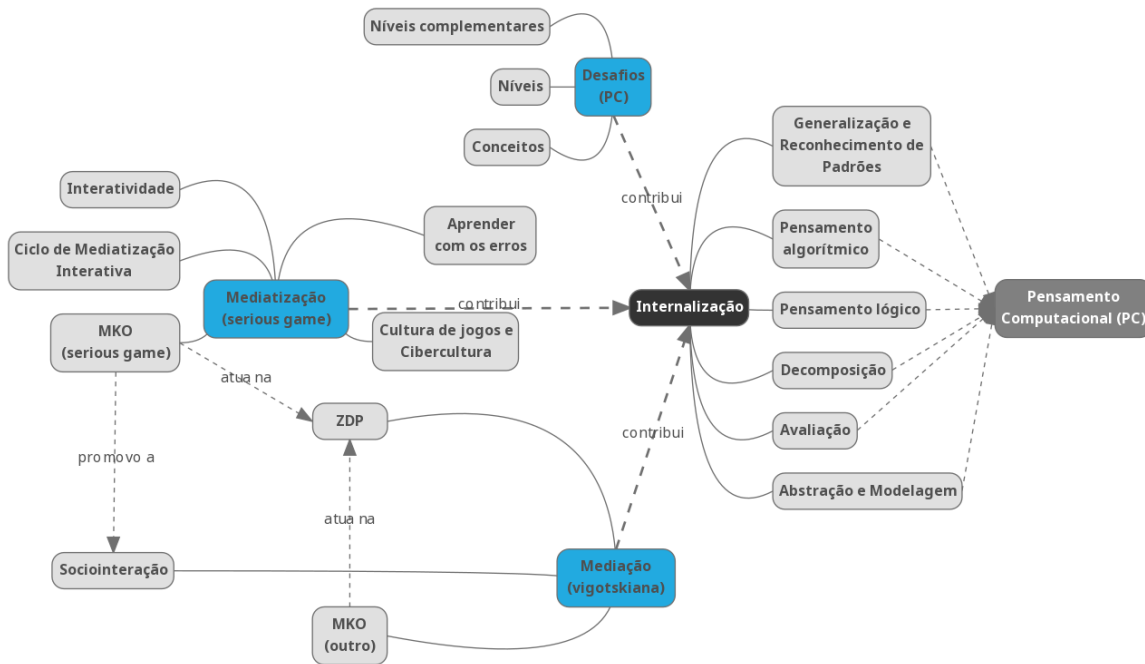
4.3.1 Como um *serious game* desenvolve o pensamento computacional?

A questão norteadora desta pesquisa buscava responder de que formas os *serious games* podem contribuir para o desenvolvimento do pensamento computacional em alunos dos anos finais do Ensino Fundamental, a partir da perspectiva *vigotskiana*. A esta altura da dissertação, em que já foram apresentados e discutidos os resultados da pesquisa, parece cabível construir uma possível resposta para tal pergunta.

Conforme sugerimos ao final da análise das produções dos alunos, na seção anterior, há indícios de internalização de conceitos do pensamento computacional, tanto em termos quantitativos quanto qualitativos, e essa internalização é mediatizada pelo jogo *CodeCombat*. Portanto, assumimos que *serious games* desse tipo **contribuem para o desenvolvimento do pensamento computacional** – uma intuição que motivou a realização desta pesquisa. Desse modo, o foco da resposta à questão de pesquisa será em *como* os jogos contribuem e não em *se*.

¹⁴ A partir dessa nova avaliação, percebemos como Donalds, um dos alunos de maior destaque ao longo da oficina ao lado de Knight, teve um aproveitamento qualitativamente melhor do que o de Ermac. No outro modelo, que considerava apenas aspectos quantitativos, Ermac aparentava um melhor aproveitamento. Em tempo, ressaltamos que não há pretensão de avaliar ou discutir o desempenho ou aproveitamento individual de nenhum dos alunos.

Figura 21 – Relações entre as categorias emergentes e os norteadores teóricos da pesquisa



Fonte: elaborado pelo autor

Nesse sentido, com base nos norteadores provenientes do quadro teórico e nas categorias emergentes da ATD, em conjunto com as evidências obtidas na análise das produções, defendemos que, por si só, um *serious game* contribui para o desenvolvimento do pensamento computacional por meio da **mediatização** e dos **desafios** que propõe aos jogadores. Não obstante, os achados da pesquisa corroboram a noção de que o potencial mediador de um *serious game* pode ser mais bem explorado quando aliado à **mediação**. O mapa mental ilustrado na Figura 21 é uma tentativa de explicar visualmente as relações que culminaram nesta resposta.

A partir desse mapa, pudemos visualizar como se relacionam as características intrínsecas da mediatização, dos desafios e da mediação, categorias que contribuem para a internalização de conceitos do pensamento computacional. Contudo, para responder à questão de pesquisa de forma aprofundada, é necessário mais do que apenas apresentar as relações entre as categorias; devemos explicá-las. Para tanto, serão retomadas questões-chave referentes a cada uma das categorias e suas características principais.

No que se refere à **mediatização**, destacamos a interatividade (CMI), a possibilidade de aprender com os erros e as características ciberculturais. Os aspectos relacionados à interatividade – como o *feedback* instantâneo – possibilitam a verificação de hipóteses e a resolução de problemas de uma forma rápida e dinâmica, o que potencializa a aprendizagem da linguagem e dos desafios do jogo por meio da exploração. Tais aspectos constituem o CMI, ciclo que desempenha um papel crucial no desenvolvimento do pen-

samento computacional, dada a natureza dos desafios de *CodeCombat*, o *serious game* pesquisado.

A possibilidade de aprender com os erros é outra característica importante da mediatização. Por conta dos fracassos ao longo da oficina, os jogadores passaram por vários CMI, isto é, vários ciclos de identificação de problemas e interpretação causal de eventos (pensar), elaboração e implementação de soluções (criar), além de verificação de hipóteses (avaliar). Em outras palavras, ao errar e tentar novamente, os jogadores tiveram mais oportunidades de aprender. Ainda no que concerne a mediatização, percebe-se que características ciberculturais como a linguagem, a desterritorialização e a disponibilidade online de *CodeCombat* também contribuem para a internalização e o desenvolvimento do pensamento computacional. Outrossim, ressaltamos que a mediatização do *serious game* potencializa o processo de internalização, pois facilita a aquisição dos signos necessários para a realização de atividades relacionadas ao pensamento computacional, atuando quase como um MKO, promovendo a sociointeração, que permite a aprendizagem por meio da imitação, mobiliza a ZDP e ensina a colaborar com o outro.

Ao lado da mediatização, o **desafio** é uma das formas pelas quais um *serious game* como *CodeCombat* contribui para o desenvolvimento do pensamento computacional. Isso ocorre, pois, seus desafios foram concebidos e elaborados com base nesse objeto de conhecimento específico, ou seja, seus desafios buscam mediar o acesso a conceitos como pensamento algorítmico e lógico, decomposição, generalização, reconhecimento de padrões, abstração, modelagem e avaliação. Desta forma, até mesmo o mais simples dos desafios de *CodeCombat* emprega a maioria dos conceitos base do pensamento computacional, o que corrobora a noção de que os jogos costumam apresentar aos jogadores problemas complexos e holísticos (GEE, 2003). Na contramão de algumas práticas pedagógicas contemporâneas, que propõem aos alunos “problemas curtos e solucionáveis onde todas as informações necessárias são apresentadas à sua frente” (SQUIRE, 2005, tradução do autor)¹⁵, *CodeCombat* convida os jogadores a superar desafios que muitas vezes apresentam alto nível de abstração, exigindo múltiplos CMI.

Finalmente, destacamos a **mediação**. Aliada à mediatização e aos desafios do *serious game*, a mediação *vigotskiana* contribui para o desenvolvimento do pensamento computacional ao fortalecer a compreensão e a atribuição de sentido do aluno em relação aos desafios do jogo, possibilitando que ele internalize os conceitos utilizados em suas soluções. Isso se deve principalmente à sociointeração e à capacidade do MKO – um professor ou aluno mais experiente – de atuar na ZDP do aluno, algo que o *serious game* por si mesmo não é capaz.

¹⁵ “short, solvable problems where all of the information needed is laid out in front of them”

4.3.2 Algumas observações necessárias

Antes de encerrarmos a discussão, cabem algumas observações pontuais – talvez até desconexas – acerca do estudo de caso como um todo. Pelo que presenciamos na oficina, o *serious game CodeCombat* parece estar alinhado ao contexto socio-histórico-cibercultural atual, pois está disponível em um ambiente online, acessível de qualquer lugar do mundo e a qualquer hora, seja nas escolas ou no conforto de uma casa. Não obstante, uma ressalva pode ser feita em relação à interface gráfica do jogo, que foi pensada exclusivamente para plataformas *desktop*¹⁶, não sendo compatível com navegadores de dispositivos móveis. Essa questão pode ser considerada uma das limitações do jogo, dado que os poucos problemas de usabilidade ocorridos durante os encontros estavam relacionados ao uso de atalhos do teclado, circunstância que pode ter relação com o fato de que nem todos os alunos possuíam computadores *desktop* ou *laptops* em suas casas. O mesmo não se pode dizer dos dispositivos móveis: todos, literalmente, possuíam um *smartphone*. Aliás, não houve um dia sequer em que os alunos não estivessem esperando o início da oficina à porta da escola jogando em seus *smartphones* (quase sempre um jogo chamado *Free Fire*)! Isso revela que *CodeCombat* poderia ter sido ainda mais aceito pelos alunos caso possuísse uma versão para plataformas *mobile* (um aplicativo para Android, iPhone etc.) ou pelo menos uma interface responsiva, que se adaptasse ao dispositivo do usuário.

Outra limitação observada é que uma parte de *CodeCombat* estava em inglês, como os recursos de vídeo, importantes para explicar o funcionamento do jogo e conceitos como *loop* e variável, que não possuíam legendas¹⁷. Assim, percebemos como é importante que o professor conheça bem o *serious game* (ou outra mídia interativa) antes de escolhê-lo como elemento mediador de suas práticas pedagógicas¹⁸.

Conhecer o contexto de utilização também é fator determinante do potencial mediador de um *serious game*: se o propósito de um jogo estiver alinhado à proposta de uma disciplina ou se ele cria a possibilidade de novas disciplinas, é provável que sua utilização seja significativa; caso contrário, o jogo servirá apenas como chamariz, como atrativo tecnológico. Isso se pode dizer da abordagem teórico-epistemológica contida num *serious game*: um professor *behaviorista* achará pouco proveitosa a utilização de um jogo construtivista e vice-versa. Outra questão relacionada ao contexto é que o ato de jogar não agrada a todos, nem mesmo aos jovens, e não há um único jogo que agrada a todos (SQUIRE, 2005). No caso desta pesquisa, entretanto, não se pôde perceber tal fenômeno provavelmente porque participaram apenas alunos com afinidade por jogos digitais.

¹⁶ Leia-se computadores de mesa e *laptops*

¹⁷ O professor chegou a fazer tradução simultânea dos vídeos para a língua portuguesa, mas a experiência não se mostrou frutífera (os alunos começaram a perder o interesse no assunto).

¹⁸ A escolha de *CodeCombat*, por exemplo, foi feita mediante várias horas de experiência de jogo por parte do professor. Logo, havia conhecimento sobre a limitação dos vídeos em inglês sem legendas. Por outro lado, sabíamos que grande parte do jogo estava traduzido para a língua portuguesa. De qualquer forma, o evidente potencial mediador do *serious game* garantiu sua escolha.

Saindo do âmbito do *serious game*, uma reflexão acerca da mediação do professor ao longo da oficina: é possível argumentar que a mediação do professor não complementou a mediatização do jogo da forma esperada. Em vez de convidar os alunos a relacionarem os desafios resolvidos com cenários reais de resolução de problemas, o professor buscou contribuir para que os alunos raciocinassem dentro do contexto do próprio jogo, esperando que eles transpusessem o conhecimento obtido dentro do mundo do jogo para a vida real por conta própria. Talvez isso seja verdade. Porém, por que os desafios solucionados no jogo não se relacionariam com a realidade? Se não estão, por que problemas matemáticos, de lógica ou de sintaxe da língua portuguesa estariam? Não faz sentido. Em outra conjuntura, mas de forma igualmente aplicável nesta argumentação, Vigotski defende que

Se aprendemos datilografia, na estrutura da nossa consciência pode não haver nenhuma mudança. Mas se aprendemos, digamos, um novo método de pensamento, um novo tipo de estruturas, isto nos dá a possibilidade não só de desenvolver a mesma atividade que fora objeto de aprendizagem imediata mas nos dá muito mais: dá a possibilidade de ir além dos limites daqueles resultados imediatos que a aprendizagem conduziu (VIGOTSKI, 2009, p. 303).

Nesse sentido, um segundo questionamento que pode ser levantado acerca dos achados da oficina é quanto à validade dos indícios de internalização, isto é, quanto à legitimidade da afirmação de que os *serious games* contribuem para o desenvolvimento do pensamento computacional. Com base no aporte teórico *vigotskiano*, entendemos que aprendizagem e desenvolvimento não são coincidentes, o que significa que nem tudo que o jogador aprende durante a experiência de jogo resulta em desenvolvimento imediato. Porém, tais aprendizagens podem ser as pedras fundamentais do desenvolvimento de diversas outras habilidades e competências no futuro.

A aprendizagem pode produzir mais no desenvolvimento que aquilo que contém em seus resultados imediatos. Aplicada a um ponto no campo do pensamento infantil, ela se modifica e refaz muitos outros pontos. No desenvolvimento ela pode surtir efeitos de longo alcance e não só aqueles de alcance imediato. Consequentemente, a aprendizagem pode ir não só atrás do desenvolvimento, não só passo a passo com ele, mas pode superá-lo, projetando-o para a frente e suscitando nele novas formações (VIGOTSKI, 2009, p. 304).

Com base nisso, argumentamos que os processos de internalização apenas iniciaram na oficina, desencadeados a partir das ações propostas em seu decorrer, por meio das interações entre alunos, *serious game* e professor-mediador. Logo, é compreensível – e até esperado – que os sujeitos da pesquisa não demonstrem de imediato um desenvolvimento decorrente das atividades realizadas no *serious game*. Eles certamente não saíram de lá programadores, mas podem vir a tornar-se, pois uma porta lhes foi aberta, um caminho lhes foi mostrado. E não necessariamente programadores devem eles se tornarem, pois o

pensamento computacional não se resume a isso. Podem se tornar engenheiros, médicos, educadores, contadores, analistas, investidores, que sabem pensar computacionalmente. Afinal, a aprendizagem não deve ser avaliada como produto, mas como processo.

4.4 Uma proposta de prática pedagógica com *serious games*

Com base no que foi aprendido durante esta pesquisa, elaboramos uma prática pedagógica relacionada ao pensamento computacional cuja mediatização ocorre por meio do *serious game CodeCombat*. Antes de apresentá-la, todavia, cabe destacarmos algumas das diretrizes que foram seguidas em sua elaboração:

- impregnar-se com os conceitos de pensamento computacional – ou qualquer que seja o objeto de conhecimento estudado;
- conhecer o contexto histórico-socio-cultural em que a prática será realizada;
- conhecer o *serious game* (ou outra mídia interativa) antes de escolhê-lo como elemento mediador da prática pedagógica. É um dever jogá-lo, mas não é necessário concluí-lo, tampouco masterizá-lo. Deve ser jogado o suficiente para que se conheça as regras, as mecânicas e as dinâmicas, assim como seu potencial pedagógico;
- saber se o *serious game* propõe claramente uma ementa ou se é necessária uma organização prévia dos assuntos e desafios a serem abordados. A partir disso, deve-se estruturar um cronograma com base na duração do jogo, prevendo também faltas de alunos, recapitulações e outros imprevistos, assim como a possibilidade de a turma avançar mais rapidamente do que o esperado.
- verificar se o jogo registra as ações do jogador de alguma forma, para posterior avaliação ou estudo de caso. Também deve-se levar em consideração a forma em que será feita a avaliação (ex: análise das produções), para que seja possível organizar-se de acordo para registrar as produções do sujeito, caso o jogo não o faça automaticamente.
- propor atividades fora do ambiente do jogo, mas que também estejam relacionadas objeto de conhecimento:
 - *atividades pré-jogo*: contextualização dos assuntos abordados no contexto do jogo (ex: o que é um algoritmo, o que é e onde se utiliza programação, etc.);
 - *atividades pós-jogo*: tarefas relacionadas aos assuntos e conceitos já trabalhados no jogo (ex: programação em quadros de papel).
- se necessário, estruturar um sistema avaliativo com base nos níveis do *serious game* e nas atividades pré e pós-jogo, tal como aquele apresentado na análise das produções,

que descreve os conceitos trabalhados em cada nível do *serious game* e qual a sua relevância para a aprendizagem. Devemos considerar os pressupostos *vigotskianos*: aprendizagem é um processo, não um produto; e aquilo que o sujeito consegue realizar em cooperação também deve ser levado em conta. Por fim, é importante atentar principalmente para o aspecto qualitativo das produções dos alunos.

Adiantamos que não será reapresentada uma proposta de utilização do *serious game CodeCombat*, pois isso seria repetitivo. Ao longo deste estudo de caso, percebemos algo observado há certo tempo por Gee (2003): os jogadores exploram, testam hipóteses, assumem riscos e enxergam erros como oportunidades para reflexão, aprendizagem e progresso; eles agem como solucionadores *ativos* de problemas. Além disso, evidenciamos seu anseio por questionar, criar e imaginar novas possibilidades para o jogo. Levando isso em consideração, elaboramos uma proposta de prática pedagógica em que é possível imaginar, criar, explorar e resolver problemas, uma atividade *offline*, pós-jogo, mas que não se distancia do *serious game* ou do pensamento computacional:

- **Descrição:** criar um nível imaginário para o jogo *CodeCombat*;
- **Contexto:** atividade pós-jogo (*CodeCombat* ou outro *serious game* relacionado ao pensamento computacional);
- **Mídia:** *offline* (papel, lápis e borracha);
- **Participantes:** Grupos de 3-4 alunos;
- **Duração:** 1 hora + 30 min × número de grupos;
- **Detalhamento da proposta:** a atividade consiste em duas partes distintas:
 1. Criação de um novo nível para o jogo *CodeCombat*:
 - o nível pode conter todas as mecânicas existentes no jogo: movimentação, ataque, inimigos, portas, palavras-chave, *loops*, itens mágicos, variáveis etc;
 - é permitido propor novas regras ou alterar as regras existentes, ou seja, pode-se criar novos golpes para o herói, novos tipos de inimigo, novos comandos, novos itens mágicos ou até mesmo criar novos conceitos, como portais, baús de tesouro etc.
 - o mapa do nível deve ser desenhado em folhas de papel, na mesma perspectiva de *CodeCombat* (visão 2D *top down*);
 - o nível proposto deve ter regras claras: o jogador deve cumprir um ou mais objetivos para completar o nível e isso deve ser comunicado por escrito junto ao desenho do nível, tal como ocorre em *CodeCombat* (ex: pegar o baú do tesouro, derrotar os inimigos e escapar do labirinto);

- comandos novos devem ser listados junto aos objetivos do nível, para que o jogador saiba para que eles servem e como utilizá-los.
2. Jogar os níveis criados pelos outros grupos:
- para cada grupo, deve-se jogar o nível construído por ele. Isso também se aplica ao grupo criador da proposta, pois é necessário testar se o nível proposto atende aos requisitos mínimos esperados (especificados na descrição da primeira parte); em outras palavras, para cada desafio, o grupo deverá escrever um algoritmo que o solucione, utilizando os comandos disponíveis;
 - os grupos podem interagir entre si para resolver os desafios uns dos outros;
 - o grupo que criou um nível que está sendo jogado por outro grupo deve ajudar na verificação da solução.
- **Preparação:**
 - explicar a proposta aos alunos;
 - destacar que o nível criado deve possuir instruções e regras claras para que os demais grupos consigam jogá-lo;
 - orientá-los a formar grupos de 3-4 integrantes (ou agrupá-los de acordo com algum critério, como um MKO por grupo, por exemplo).
 - **Acompanhamento:**
 - o professor-mediador precisa acompanhar os grupos durante a primeira parte da atividade principalmente para assegurar que a proposta foi compreendida;
 - na medida do possível, é importante controlar o andamento da construção dos níveis, para que posteriormente os grupos tenham tempo para jogar;
 - na segunda parte, é importante ficar atento para oportunidades de mediação, garantindo que os integrantes do grupo compreendam as regras de cada nível e esclarecendo as dúvidas.
 - **Fechamento:**
 - dar *feedback* aos alunos sobre os níveis construídos, convidando-os para que também deem os seus;
 - questionar aos alunos quais momentos da atividade foram interessantes e como ela se compara à experiência de jogo;
 - comparar aquilo que foi construído pelos grupos ao que foi experimentado no ambiente do jogo e, se possível, relacionar aos conceitos do pensamento computacional e a contextos aplicáveis na vida real, explicitando por que pode ser importante estudar tal objeto de conhecimento.

- **Avaliação:**

- verificar se o grupo construiu um nível para o jogo *CodeCombat*, atentando para aspectos como imaginação, criatividade e complexidade, bem como indícios de internalização de conceitos do pensamento computacional. Por exemplo, uma proposta que envolva o uso de *loops* pode indicar que houve generalização e reconhecimento de padrões da noção de repetição; ao mesmo tempo, a criação de uma nova mecânica (ex: um novo poder do personagem) pode ser vista como algo criativo;
- verificar se o grupo descreveu claramente as regras do nível elaborado, de modo que os demais grupos soubessem quais ações eram permitidas, isto é, quais comandos poderiam executar e como saberiam testar por condições de vitória ou derrota;
- verificar se o grupo conseguiu (ou pelo menos se propôs a) resolver os níveis construídos pelos outros grupos;
- considerar também a participação individual de cada um, tanto na elaboração do nível quanto na resolução dos níveis propostos pelos outros grupos.

Finalmente, destaca-se que esta proposta de prática pedagógica é apenas uma alternativa, uma tentativa de síntese de tudo aquilo que foi aprendido ao longo de dois anos de pesquisa. Portanto, ela ainda não serve como um *framework*. Não obstante, esperamos que ela possa servir como inspiração e como base para a elaboração de propostas mais robustas e adaptadas ao contexto. Ressaltamos também que a abordagem *vigotskiana* pode ser ampliada para além de contextos onde *CodeCombat* ou outro *serious game* está presente. Propomos que esse resultado possa ser transposto a ambientes onde existam outros recursos, como objetos de aprendizagem, jogos de entretenimento e ferramentas no estilo *Scratch*, até mesmo para atividades de computação desplugada.

5 Considerações finais

Foi uma longa jornada, repleta de idas e vindas, de altos e baixos, e, sobretudo, de aprendizagens. Sendo assim, é um momento propício para avaliar se os objetivos propostos foram alcançados, de que formas esta pesquisa contribui para a comunidade científica e quais são suas limitações, assim como quais são suas perspectivas para trabalhos futuros.

Em se tratando dos objetivos específicos, podemos dizer que eles foram alcançados – afinal, conforme mencionado no capítulo do método, são eles que apontam as direções que o pesquisador deve seguir para conduzir a pesquisa. O primeiro objetivo específico se referia à construção de um referencial teórico que contemplasse os conceitos de educação, *serious games*, pensamento computacional, e a abordagem *vigotskiana* de aprendizagem e desenvolvimento. Esse objetivo foi concluído ainda no primeiro ano de projeto e serviu como base para a pesquisa como um todo. Do referencial teórico construído, foram gerados os norteadores teóricos, cumprindo-se assim o segundo objetivo específico e consolidando a fundamentação teórica da dissertação. O terceiro objetivo específico, por sua vez, consistia na realização de uma oficina de Introdução à Ciência da Computação mediada pelo *serious game CodeCombat*. Os encontros da oficina permitiram-nos conhecer melhor o contexto estudado e os aspectos práticos da educação. Ao mesmo tempo, gerou-nos o *corpus*, a partir do qual realizamos o quarto e último objetivo específico, a análise.

No que diz respeito ao objetivo geral, a identificação das contribuições dos *serious games* para o desenvolvimento do pensamento computacional e a elaboração de uma proposta de prática pedagógica, podemos dizer que ele foi alcançado. Sua primeira parte estava fortemente vinculada à questão de pesquisa. Como esta foi respondida, é plausível afirmar que a primeira parte do objetivo também foi concluída. A segunda parte, todavia, não estava diretamente relacionada a nenhum objetivo específico, isto é, foi necessário um esforço independente para que ela fosse realizada. A elaboração da proposta de prática pedagógica foi uma das últimas atividades realizadas na dissertação, pois consistia num exercício de síntese de tudo que ocorrera no estudo de caso. Arriscamos afirmar que essa segunda parte do objetivo geral também foi concluída, por mais simples que a prática pedagógica proposta possa parecer.

Nesse sentido, cabe dizer que o cumprimento desses objetivos pode ter gerado algumas contribuições para a comunidade científica na área da pesquisa em jogos e pensamento computacional, em especial o capítulo em que apresentamos e discutimos os resultados. A ideia de associar jogos digitais e pensamento computacional e levá-los às escolas não é novidade no meio acadêmico. Não obstante, esta dissertação apresenta novas lentes teóricas para a discussão desse tema, inovando principalmente no que se refere ao

tipo de jogo (*serious games*) e à abordagem *vigotskiana*. Por conta disso, as análises e interpretações do estudo de caso podem servir como referência ou contraponto para outros pesquisadores da área. Além disso, destacamos os construtos teóricos da “mediatização” e o CMI (Ciclo de Mediatização Interativa), que podem ser ampliados e aprofundados em novas pesquisas. No que toca à proposta de prática pedagógica, por ser incipiente, ela ainda não serve como um modelo científico, um *framework*, mas pode vir a servir como base para a elaboração de propostas mais robustas e como inspiração para professores que têm interesse em ousar e levar não somente *games* como também o pensamento computacional às salas de aula.

Isso posto, não seria honesto deixar de mencionar as limitações que este trabalho possui. Pelo menos duas importantes limitações nos ficaram evidentes, mas pode haver muitas outras. A primeira refere-se aos achados da pesquisa: não foram abordadas questões relevantes como motivação, imersão, diversão e fluxo, elementos que costumam ser problematizados em pesquisas relacionadas aos jogos digitais em ambientes de aprendizagem. Inicialmente, houve a intenção de abordar tais questões na categoria “mediatização”, visto que elas foram percebidas durante os procedimentos de análise. Contudo, o texto resultante de sua análise não foi satisfatório e, conseqüentemente, foi cortado da versão final da dissertação. A segunda limitação, por sua vez, está mais relacionada ao método: nenhuma menina participou da oficina. Diferentemente do piloto, em que houve interesse por parte de três meninas, apesar de nenhuma ter participado de todos os encontros, a oficina oficial sequer atraiu o interesse das meninas do 7º e 8º ano. Portanto, os resultados e conclusões deste estudo de caso se limitam por se distanciarem do contexto real do fenômeno.

De qualquer forma, ressaltamos que o tema desta pesquisa não se esgotou por aqui, pois existem perspectivas para trabalhos futuros. Dentre elas estão algumas “pontas soltas”, questões que não foram aprofundadas nesta dissertação devido ao escopo, e até mesmo novas questões norteadoras, capazes de gerar novas pesquisas. Algumas das perspectivas incluem:

- replicar a mesma pesquisa, mas com sujeitos de ambos os sexos, para que o contexto do estudo de caso seja mais verossímil;
- estender a análise do *corpus*, debruçando-se sobre questões que não foram aprofundadas, como o desenvolvimento de conceitos científicos (no sentido *vigotskiano*) e a motivação, por exemplo;
- aprofundar uma possível relação entre a noção *vigotskiana* de processos psicológicos superiores e as tecnologias da inteligência, no sentido cunhado por Levy, argumentando que o uso de tais tecnologias pode desenvolver novos processos ou modificar/ampliar os processos já existentes;

-
- analisar o *corpus* a partir de outro aporte teórico/outra teoria da aprendizagem, utilizando autores como Piaget, Maturana etc.;
 - realizar uma pesquisa quantitativa para corroborar ou refutar os resultados apresentados;
 - ampliar a proposta de prática pedagógica para a dimensão de um curso, por exemplo, e analisar sua aplicação;
 - realizar um estudo comparativo entre diferentes *serious games*, para verificar se *CodeCombat* realmente foi uma boa escolha para esta pesquisa e se existem alternativas de mesmo nível ou superiores para o desenvolvimento do pensamento computacional;
 - realizar um estudo comparativo utilizando jogos digitais de entretenimento e *serious games*, com o intuito de verificar se os últimos realmente são mais adequados para fins educativos.

A partir dessas perspectivas, temos a impressão quase esmagadora – e um tanto quanto socrática – de que conhecemos menos ainda do que pensávamos conhecer no início da jornada. Paradoxalmente, percebemos que algum conhecimento foi construído aqui, senão não haveria tantas possibilidades de continuação para esta pesquisa.

Finalmente, confessamos que um projeto de pesquisa que parecia fácil de executar provou-se um grande desafio ao longo destes dois anos. Hoje, a partir das vivências adquiridas, entendemos que compreender os fenômenos relacionados à educação, à linguagem e às tecnologias é mais complicado do que parece na teoria. E transformar o estado das coisas é ainda mais difícil. É provavelmente por isso que a busca por novas formas de ensinar e aprender jamais finda. Afinal, é nosso dever como pesquisadores da área da educação perseverar nessa caminhada.

Referências

ABT, C. C. *Serious games*. Lanham: University Press Of America, 1987. 176 p. Citado na página 49.

BARCELOS, T. S. *Relações entre o Pensamento Computacional e a Matemática em atividades didáticas de construção de jogos digitais*. 276 p. Tese (Doutorado) — Curso de Doutorado em Ensino de Ciências e Matemática, Universidade Cruzeiro do Sul, São Paulo, 2014. Citado 2 vezes nas páginas 48 e 55.

BEECHER, K. *Computational thinking: A beginner's guide to problem-solving and programming*. [S.l.]: Bcs, 2017. 288 p. Citado 5 vezes nas páginas 55, 56, 57, 91 e 94.

BLIKSTEIN, P. *O pensamento computacional e a reinvenção do computador na educação*. 2008. Disponível em: <http://www.blikstein.com/paulo/documents/online/ol_pensamento_computacional.html>. Acesso em: 09 jan. 2019. Citado na página 57.

BRASIL. Ministério da Educação. *Base Nacional Comum Curricular - BNCC*. 2017. Disponível em: <<http://basenacionalcomum.mec.gov.br/abase>>. Acesso em: 01 dez. 2019. Citado na página 34.

BRUMER, A. et al. A elaboração do projeto de pesquisa em ciências sociais. In: _____. *Ciências humanas: Pesquisa e método*. Porto Alegre: Ufrgs, 2008. (Graduação), p. 125–146. Citado na página 59.

CODECOMBAT. *Frequently Asked Questions*. 2020. Disponível em: <<https://br.codecombat.com/teachers/resources/faq>>. Acesso em: 22 jan. 2020. Citado na página 62.

CODECOMBAT. *Introdução à Programação em Pares*. 2020. Disponível em: <<https://br.codecombat.com/teachers/resources/pair-programming-pt-BR>>. Acesso em: 22 jan. 2020. Citado na página 65.

CODECOMBAT. *Sobre*. 2020. Disponível em: <<https://br.codecombat.com/about>>. Acesso em: 22 jan. 2020. Citado na página 141.

COLE, M. The perils of translation: A first step in reconsidering vygotsky's theory of development in relation to formal education. *Mind, Culture, And Activity*, Informa UK Limited, v. 16, n. 4, p. 291–295, out. 2009. Citado 4 vezes nas páginas 42, 46, 47 e 105.

COLE, M.; LEVITIN, K.; LURIA, A. R. *The autobiography of Alexander Luria: a dialogue with the making of mind*. New York: Psychology Press, 2005. 276 p. Citado na página 44.

COULDRY, N.; HEPP, A. Conceptualizing mediatization: Contexts, traditions, arguments. *Communication Theory*, v. 23, n. 3, p. 191–202, Aug 2013. Citado na página 73.

DJAOUTI, D. *Serious Game Design: considérations théoriques et techniques sur la création de jeux vidéo à vocation utilitaire*. 330 p. Tese (Doutorado) — Informatique, Université Toulouse Iii Paul Sabatier, Toulouse, 2011. Citado na página 50.

- DJAOUTI, D. Serious games pour l'éducation: utiliser, créer, faire créer ? *Tréma*, n. 44, p. 51–64, mai. 2016. Citado 4 vezes nas páginas 50, 52, 53 e 85.
- FARDO, M. L. *A gamificação como estratégia pedagógica: estudo de elementos dos games aplicados em processos de ensino e aprendizagem*. 104 p. Dissertação (Mestrado) — Curso de Mestrado em Educação, Universidade de Caxias do Sul, Caxias do Sul, 2013. Citado 2 vezes nas páginas 48 e 52.
- FLICK, U. *Introdução à pesquisa qualitativa*. 3. ed. Porto Alegre: Artmed, 2009. 405 p. Citado na página 59.
- GEE, J. P. *What video games have to teach us about learning and literacy*. New York: Palgrave Mcmillan, 2003. Citado 4 vezes nas páginas 35, 50, 121 e 125.
- GEE, J. P. *Why video games are good for your soul*. Victoria, Australia: Common Ground, 2005. Citado 2 vezes nas páginas 35 e 50.
- HABGOOD, M. P. J. *The Effective Integration of Digital Games and Learning Content*. 262 p. Tese (Doutorado) — University Of Nottingham, Nottingham, Reino Unido, 2007. Citado 2 vezes nas páginas 52 e 53.
- HOCHET, Y. Jeux vidéo et enseignement de l'histoire et de la géographie. In: _____. *Les jeux vidéo comme objet de recherche*. [S.l.]: L > P, 2011. (Questions Théoriques), p. 103–112. Citado na página 52.
- INSTITUTO NACIONAL DE ESTUDOS E PESQUISAS EDUCACIONAIS ANÍSIO TEIXEIRA (INEP). *Sinopses Estatísticas da Educação Superior 2016*. Brasília, 2017. Disponível em: <<http://portal.inep.gov.br/web/guest/sinopses-estatisticas-da-educacao-superior>>. Acesso em: 01 dez. 2019. Citado na página 32.
- JAQUES, R. R. *Educação e linguagem: As situações enunciativas do role-playing game (rpg) como ferramenta pedagógica de constituição da alteridade*. 104 p. Dissertação (Mestrado) — Curso de Mestrado em Educação, Universidade de Caxias do Sul, Caxias do Sul, 2016. Citado na página 48.
- KENSKI, V. M. *Educação e tecnologias: O novo ritmo da informação*. 8. ed. Campinas: Papirus, 2011. 141 p. Citado na página 40.
- LE MOS, A. Ciber-cultura-remix. In: _____. *Imagem (Ir) realidade. Comunicação e cibermídia*. Porto Alegre: Sulina, 2006. p. 53–65. Citado na página 40.
- LE MOS, A. Infraestrutura para a cultura digital. In: _____. *Cultura digital.br*. Rio de Janeiro: Beco do Azogue, 2009. p. 134–149. Citado na página 40.
- LE MOS, A. *Cibercultura: Tecnologia e vida social na cultura contemporânea*. 6. ed. Porto Alegre: Sulina, 2013. 296 p. Citado na página 39.
- LEVY, P. *Cibercultura*. 3. ed. São Paulo: Editora 34, 2010. 272 p. (Coleção TRANS). Citado 2 vezes nas páginas 39 e 40.
- LIEBERMAN, D. A. What can we learn from playing interactive games? In: _____. *Playing video games: Motives, responses, and consequences*. [S.l.]: Lawrence Erlbaum Associates Publishers, 2006. p. 379–397. Citado na página 75.

- LOOP. In: *DICTIONARY of Computing*. 5. ed. London: Bloomsbury Publishing Plc, 2004. p. 385. Citado na página 102.
- LUDKE, M.; ANDRE, M. *Pesquisa em educação: abordagens qualitativas*. 2. ed. Rio de Janeiro: Epu, 2013. 112 p. Citado na página 59.
- MALONE, T. W. Toward a theory of intrinsically motivating instruction*. *Cognitive Science*, v. 5, n. 4, p. 333–369, 1981. Citado na página 80.
- MARCUS, S. Ten human needs that education should take into account. *Theoretical Computer Science*, v. 701, p. 8–11, 2017. Citado na página 80.
- MICHAEL, D.; CHEN, S. *Serious Games: Games that educate, train, and inform*. 2. ed. Connecticut: Cengage Learning Ptr, 2005. Citado 5 vezes nas páginas 35, 49, 50, 52 e 80.
- MORAES, M. C. d. Tecendo a rede, mas com que paradigma? In: _____. *Educação a distância: fundamentos e práticas*. Campinas: Nied/unicamp, 2002. cap. 1, p. 1–25. Citado 2 vezes nas páginas 40 e 41.
- MORAES, R.; GALIAZZI, M. d. C. *Análise textual discursiva*. 2. ed. Ijuí, RS: UNIJUÍ, 2011. 223 p. (Coleção educação em ciências). Citado 4 vezes nas páginas 66, 67, 68 e 69.
- MORAES, R. M. d. *Gênese da linguagem na teoria sócio-cultural de L.S. Vygotsky*. 2010. Disponível em: <<https://pt.slideshare.net/ronnymm/gnese-do-pensamento-e-da-linguagem-l-s-vygotsky>>. Acesso em: 08 dez. 2019. Citado na página 43.
- OLIVEIRA, M. K. d. Vygotsky e o processo de formação de conceitos. In: _____. *Piaget ; Vygotsky ; Wallon: teorias psicogenéticas em discussão*. 27. ed. São Paulo: Summus Editorial, 2016. p. 23–34. Citado 2 vezes nas páginas 44 e 45.
- PAPERT, S. *A máquina das crianças: repensando a escola na era da informática*. Porto Alegre: Artmed, 2008. 224 p. Citado 2 vezes nas páginas 53 e 54.
- PAVIANI, J. *Epistemologia prática: ensino e conhecimento científico*. Caxias do Sul: Educs, 2009. 143 p. Citado na página 60.
- PAVIANI, J. *Uma introdução à filosofia*. Caxias do Sul: Educs, 2014. 107 p. Citado na página 39.
- PIVEC, M.; PIVEC, P. What do we know from research about the use of games in education? In: _____. *How are digital games used in schools? Complete results of the study*. Bruxelas, Bélgica: European Schoolnet, 2009. cap. 7, p. 121–156. Disponível em: <http://games.eun.org/upload/gis-full_report_en.pdf>. Acesso em: 1 dez. 2019. Citado 3 vezes nas páginas 51, 52 e 93.
- POLONI, L. *Aprendizagem de programação mediada por uma linguagem visual: possibilidade de desenvolvimento do pensamento computacional*. 181 p. Dissertação (Mestrado) — Programa de Pós-graduação em Educação, Universidade de Caxias do Sul, Caxias do Sul, 2018. Citado 2 vezes nas páginas 47 e 55.
- POLYA, G. *How to solve it: a new aspect of mathematical method mathematical method*. [S.l.]: Princeton University Press, 1971. Citado na página 77.

- PRENSKY, M. Digital game-based learning. *Computers In Entertainment*, v. 1, n. 1, p. 21–21, oct. 2003. Citado 3 vezes nas páginas 35, 50 e 75.
- PRESTES, Z. *Quando não é quase a mesma coisa: traduções de lev semionovitch vigotski no brasil*. Campinas: Autores Associados, 2012. 272 p. Citado 2 vezes nas páginas 42 e 45.
- RATNER, C. *Vygotsky's sociohistorical psychology and its contemporary applications*. New York: Springer Science+business Media, 1991. 368 p. (Cognition and Language). Citado 4 vezes nas páginas 42, 44, 45 e 47.
- RESNICK, M. *Let's teach kids to code*. 2012. Disponível em: <https://www.ted.com/talks/mitch_resnick_let_s_teach_kids_to_code>. Acesso em: 17 set. 2018. Citado na página 55.
- RITTERFELD, U.; CODY, M. J.; VORDERER, P. (Ed.). *Serious Games: Mechanisms and effects*. New York: Routledge, 2009. 530 p. Citado 2 vezes nas páginas 35 e 49.
- RITTERFELD, U. et al. Multimodality and interactivity: Connecting properties of serious games with educational outcomes. *Cyberpsychology & Behavior*, v. 12, n. 6, p. 691–697, 2009. Citado na página 74.
- RITTERFELD, U.; WEBER, R. Video games for entertainment and education. In: _____. *Playing Video Games: Motives, Responses and Consequences*. Mahwah, New Jersey: Lawrence Erlbaum Associates, 2006. cap. 27, p. 399–413. Citado na página 50.
- SALVADORI, P. R. *Teoria e percepção musical: práticas pedagógicas mediadas pelo earmaster*. 94 p. Dissertação (Mestrado) — Curso de Mestrado em Educação, Universidade de Caxias do Sul, Caxias do Sul, 2016. Citado 2 vezes nas páginas 47 e 48.
- SANCHEZ, E. Usage d'un jeu sérieux dans l'enseignement secondaire: Modélisation comportementale et épistémique de l'apprenant. *Revue D'intelligence Artificielle*, Lavoisier, v. 25, n. 2, p. 203–222, abr. 2011. Citado na página 51.
- SANTOS, E. A. A. d. *Contribuição para o Estudo da Aprendizagem da Matemática e da Programação em Comunidades Virtuais de Prática com Foco no Uso de Robots como Mediadores da Aprendizagem*. 212 p. Tese (Doutorado) — Curso de Doutorado em Matemática, Universidade da Madeira, Funchal, Portugal, 2012. Citado na página 47.
- SEVERGNINI, L. F.; SOARES, E. M. d. S. O serious game codecombat e o professor como mediadores da aprendizagem do pensamento computacional. In: *Anais do SBIE 2019*. [S.l.: s.n.], 2019. p. 684–693. Citado 3 vezes nas páginas 47, 48 e 52.
- SEVERGNINI, L. S. Alice e o mistério dos algoritmos: um serious game como ferramenta de aprendizagem de lógica de programação para crianças. *Renote*, Porto Alegre, v. 16, n. 1, p. 864–897, set. 2018. Disponível em: <<http://www.seer.ufgrs.br/renote/article/view/86049>>. Citado 2 vezes nas páginas 48 e 52.
- SQUIRE, K. Changing the game: What happens when video games enter the classroom? *Innovate: Journal of Online Education*, v. 1, n. 6, jan. 2005. Disponível em: <<https://nsuworks.nova.edu/innovate/vol1/iss6/5>>. Citado 3 vezes nas páginas 81, 121 e 122.

- VALENTE, J. A. Integração do pensamento computacional no currículo da educação básica: diferentes estratégias usadas e questões de formação de professores e avaliação do aluno. *Revista E-curriculum*, v. 14, n. 3, p. 864–897, set. 2016. Disponível em: <<https://revistas.pucsp.br/index.php/curriculum/article/view/29051>>. Citado 2 vezes nas páginas 34 e 57.
- VIGOTSKI, L. S. The instrumental method in psychology. Transcrição de uma palestra apresentada na Krupskaya Academy of Communist Education. 1930. Disponível em: <<https://www.marxists.org/archive/vygotsky/works/1930/instrumental.htm>>. Citado na página 43.
- VIGOTSKI, L. S. *A formação social da mente: O desenvolvimento dos processos psicológicos superiores*. 7. ed. São Paulo: Martins Fontes, 2007. 182 p. (Psicologia). Citado 16 vezes nas páginas 33, 42, 43, 45, 46, 47, 48, 49, 51, 52, 82, 83, 84, 85, 94 e 107.
- VIGOTSKI, L. S. *A construção do pensamento e da linguagem*. 2. ed. São Paulo: Martins Fontes, 2009. 520 p. Citado 6 vezes nas páginas 34, 83, 84, 106, 107 e 123.
- VYGOTSKY, L. S. *Mind in society: the development of higher psychological processes*. Cambridge, Massachusetts: Harvard University Press, 1978. Citado na página 45.
- WASTIAU, P. et al. *How are digital games used in schools?: Complete results of the study*. Bruxelas, Bélgica: European Schoolnet, 2009. 174 p. Disponível em: <http://games.eun.org/upload/gis-full_report_en.pdf>. Acesso em: 08 dez. 2019. Citado na página 52.
- WHITTON, N. Game engagement theory and adult learning. *Simulation & Gaming*, v. 42, n. 5, p. 596–609, ago. 2010. Citado 2 vezes nas páginas 51 e 75.
- WING, J. M. Computational thinking. *Communications Of The Acm*, v. 49, n. 3, p. 33–35, mar. 2006. Citado 4 vezes nas páginas 34, 53, 54 e 56.
- WING, J. M. *Computational Thinking Benefits Society*. 2014. Disponível em: <<http://socialissues.cs.toronto.edu/index.html>>. Acesso em: 17 set. 2018. Citado 3 vezes nas páginas 55, 56 e 57.
- YASNITSKY, A.; VEER, R. v. D. (Ed.). *Revisionist Revolution in Vygotsky Studies: The state of the art*. London And New York: Routledge, 2015. Citado na página 42.
- YIN, R. K. *Estudo de caso: planejamento e métodos*. 4. ed. Porto Alegre: Bookman, 2010. 248 p. Citado na página 61.
- YONG, S. T.; GATES, P.; CHAN, A. T.-Y. Similarities and differences in learning of metacognitive skills: Computer games versus mathematics education. *International Journal of Game-Based Learning*, v. 9, n. 1, p. 1–14, 2019. Citado na página 80.
- ZIMMERMAN, E. Narrative, interactivity, play, and games: Four naughty concepts in need of discipline. In: _____. *First Person: New media as story, performance, and game*. [S.l.]: MIT Press, 2004. p. 154–164. Citado 2 vezes nas páginas 67 e 74.

Apêndices

APÊNDICE A – O *serious game* *CodeCombat* em uma casca de noz

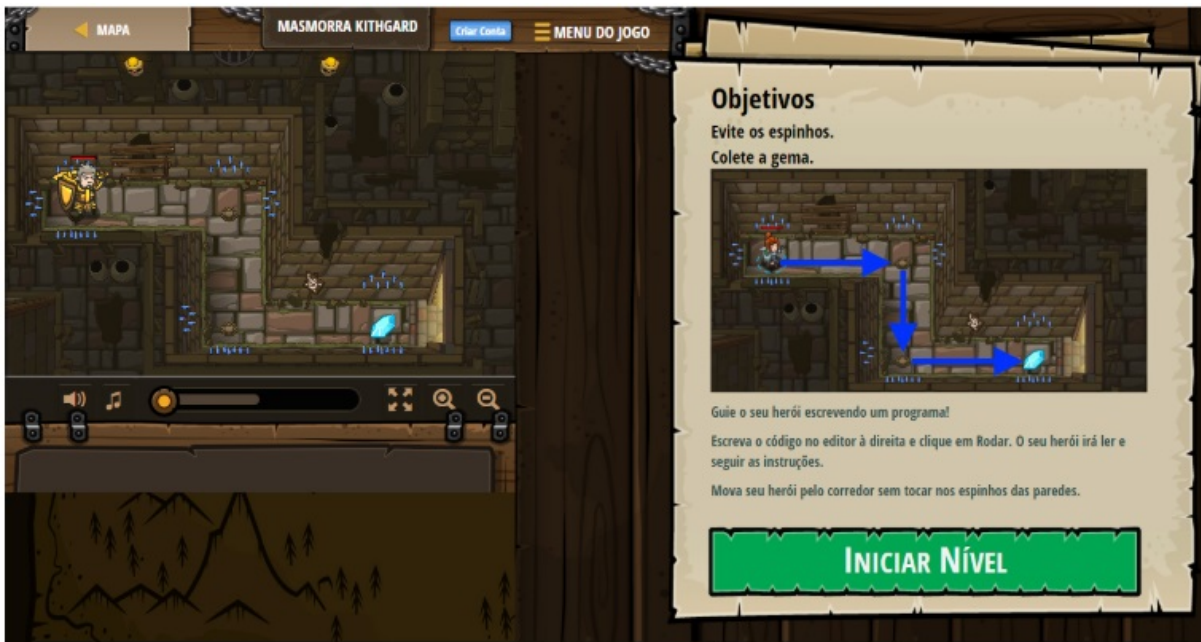
CodeCombat é um jogo de aventura RPG (*role-playing game* ou jogo de interpretação de papéis) em que o jogador aprende a programar *de verdade* enquanto enfrenta desafios como um herói das histórias de literatura fantástica. Aprender a programar de verdade, neste caso, remete ao fato de que o código produzido pelos jogadores é digitado, tal como o é na realidade, diferentemente das plataformas *Scratch* e *Code.org*, em que o código é produzido através de blocos de arrastar e soltar (*drag and drop*).

A equipe de desenvolvimento de *CodeCombat* entende que *programação é mágica*, uma habilidade de criar coisas a partir da imaginação. Reconhecendo essa percepção, tal equipe desenvolveu – e continua a desenvolver – um *serious game* para “dar aos alunos a sensação de poder de um mago na ponta de seus dedos enquanto digitam códigos” (CODECOMBAT, 2020c). Levando isso em consideração, pensávamos que esse jogo, em conjunto com a mediação de um professor, possibilitaria a realização de uma oficina que levasse o *serious game* ao limite de suas potencialidades, oferecendo oportunidades de aprendizagem de uma forma lúdica.

Figura 22 – Ambientação do jogo *CodeCombat* e exemplo de solução (à direita)



Fonte: <https://br.codecombat.com/about>

Figura 23 – Objetivos do primeiro nível do jogo *CodeCombat*

Fonte: elaborado pelo autor

Em *CodeCombat*, para derrotar um inimigo, não basta que o jogador pense em uma estratégia e a execute em um “pressionar de botões”. Para fazê-lo, o jogador precisa traduzir sua estratégia em uma linguagem que o jogo compreenda – linguagem essa que é representada por comandos próprios do jogo, que ao serem executados pelo personagem podem levar à “derrota do inimigo”. Isso significa que, para cumprir os objetivos de cada fase do jogo, os jogadores devem escrever pequenos algoritmos, sequências de instruções que descrevem a solução de um problema, de forma não-ambígua, em uma linguagem de programação. Em outras palavras, o herói do jogo realiza as ações necessárias para completar um determinado nível se e somente se for instruído corretamente pelo jogador. Caso contrário, o jogador precisa revisar e corrigir sua solução. Como se há de argumentar mais adiante, ocasiões como essa, em que um aluno apresenta dificuldades para resolver um desafio, podem ser oportunidades para atuar em sua ZDP e ajudá-lo a elucidar uma solução por meio de problematizações, pistas ou outras táticas que forneçam os subsídios necessários.

A escolha do jogo *CodeCombat* baseou-se no pressuposto de que as ações nele tomadas desencadeariam processos internos associados ao pensamento lógico-algorítmico, à decomposição, à generalização, ao reconhecimento de padrões e à abstração, que, por sua vez, fazem parte do pensamento computacional. Uma experiência de jogo como essa parecia estar fortemente vinculada ao que seja programação, e com o pensamento computacional, num nível de abstração aparentemente adequado à faixa etária dos sujeitos da pesquisa.

Finalmente, *CodeCombat* foi selecionado por se esperar que sua experiência de jogo, articulada com as intervenções de um mediador, gradativamente tornasse possível ao

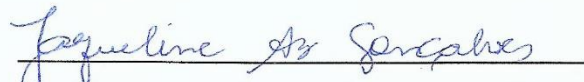
jogador operar mentalmente (elaborar estratégias) sem a presença dos referentes concretos (cenário do jogo) – fenômeno que, no aporte teórico de Vigotski, se conceitua como *internalização*. No momento de planejamento do estudo de caso, isso já parecia uma tendência, uma vez que tanto o *serious game* (através da própria linguagem do jogo) quanto o mediador (por meio de diálogos, problematizações e questionamentos) estariam fornecendo representações simbólicas para tais operações, caracterizando uma mediação *vigotskiana*.

APÊNDICE B – Termo de Anuência Institucional

TERMO DE ANUÊNCIA

Declaramos para os devidos fins que estamos de acordo com a execução do projeto de pesquisa intitulado "SERIOUS GAMES COMO ELEMENTO MEDIADOR DO DESENVOLVIMENTO DO PENSAMENTO COMPUTACIONAL", sob a coordenação e a responsabilidade de Luís Filipe Severgnini, mestrando do Programa de Pós-Graduação em Educação, da Área do Conhecimento de Humanidades, da Universidade de Caxias do Sul, o qual terá o apoio desta Instituição.

Farroupilha, 26 de dezembro de 2018.



Jaqueline de Albuquerque Borges Gonçalves – Diretora Escolar
(carimbar)

Jaqueline de A. Borges Gonçalves
Diretora
Portaria 1.056/2015

Esc. Mun. de Ensino Fundamental
ZELINDA RODOLFO PESSIN
R. Carlos Buscaino, 184, Centenário
95177-204 - Farroupilha/RS
54 3261-2818
escolazelindapessin@farroupilha.rs.gov.br

APÊNDICE C – Termo de Consentimento Livre e Esclarecido

APÊNDICE B – TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

O termo de consentimento livre e esclarecido será entregue a cada um dos responsáveis dos participantes da oficina, os quais deverão assinar e rubricar cada uma das vias, autorizando a participação da(s) criança(a) sob sua responsabilidade na pesquisa. Uma das vias será retida pelo pesquisador.

Projeto de pesquisa: jogos eletrônicos (*serious games*) como elemento mediador do desenvolvimento do pensamento computacional.

Pesquisador responsável: Luís Filipe Severgnini

Orientadora: Dra. Eliana Maria do Sacramento Soares

Instituição: Universidade de Caxias do Sul

Objetivo: Identificar as contribuições dos jogos eletrônicos (*serious games*) no desenvolvimento do pensamento computacional e, a partir disso, elaborar uma proposta de prática pedagógica em que jogos desse gênero participem como elementos mediadores do desenvolvimento de tal forma de pensamento.

Benefícios: O principal benefício desta pesquisa é produzir novos conhecimentos relacionados ao desenvolvimento do pensamento computacional, uma abordagem para solução de problemas que envolve o uso de um conjunto de práticas e princípios da ciência da computação para formular soluções executáveis por um computador, não exclusiva para programadores, e aplicável em uma variedade diversificada de campos. Compreender melhor esse assunto pode contribuir para que um número maior de estudantes sejam criadores de tecnologias em vez de apenas usuários. Ao mesmo tempo, serão investigadas as contribuições dos jogos eletrônicos (*serious games*) nesse processo, o que pode abrir espaço para uma forma mais lúdica, prazerosa e motivadora de aprender o pensamento computacional.

Riscos: Entende-se que os riscos da pesquisa sejam mínimos, pois ela não aborda nenhum assunto moralmente controverso e tampouco prevê atividades físicas que possam causar lesões e/ou traumas. No entanto, eventualmente, as atividades podem gerar algum desconforto, *stress*, cansaço ou tédio nos participantes. Por isso, estes podem desistir de participar das atividades – ou até mesmo da pesquisa como um todo – a qualquer momento, sem nenhum prejuízo de sua parte. A seguir, descreve-se com detalhamento os riscos envolvidos na pesquisa:

- **DESCONFORTO, STRESS OU CANSAÇO.** Prevenção: esclarecer que as atividades da oficina não têm caráter avaliativo e não estão vinculadas às atividades regulares e, por isso, não há necessidade para nervosismo ou medo de errar; motivar os sujeitos a buscar a diversão em primeiro lugar. Reparação: aumentar o tempo de cada atividade; incentivar os sujeitos a trabalharem em conjunto; acompanhar a realização da atividade do sujeito; interromper a realização da atividade.
- **TÉDIO.** Prevenção: acompanhar os movimentos dos sujeitos com o intuito de perceber frustrações com o jogo, seja por causa do nível de dificuldade ou por desinteresse no jogo em si. Reparação: em caso de algum desafio se tornar frustrante, causando ansiedade nos sujeitos, fornecer pistas e realizar problematizações de modo a possibilitar que o desafio seja superado. Caso os jogadores achem o jogo desinteressante, existe a possibilidade de substituir o jogo.
- **FRUSTRAÇÃO EM NÃO CONSEGUIR CUMPRIR AS ATIVIDADES PROPOSTAS.** Prevenção: esclarecer as atividades propostas; estabelecer um ambiente acolhedor que incentive a tentativa e o erro por parte dos alunos, bem como a elaboração de perguntas quando houver dúvidas ou curiosidades. Reparação: explicar que não há problemas em errar, enfatizando que o erro faz parte da aprendizagem; caso o sujeito esteja disposto a continuar, incentivar nova tentativa; em caso de repetidas frustrações, adequar a atividade para que atenda o nível de desenvolvimento do sujeito.

- **CONSTRANGIMENTOS DE EXPOSIÇÃO.** Prevenção: esclarecer aos sujeitos que ninguém além do pesquisador terá acesso às gravações da oficina; esclarecer que as gravações estarão armazenadas de maneira segura; em caso de ausência de acordo mútuo dos sujeitos, suspender as gravações. Reparação: se os sujeitos demonstrarem desconforto ou insatisfação com a exposição gerada pelas filmagens, estas serão suspensas.
- **CONFLITO ENTRE OS PARTICIPANTES.** Prevenção: incentivar a cooperação entre os participantes e, em caso de existir competições ou rivalidades no grupo, encorajar que a competição seja salutar. Reparação: retomar as ações preventivas. Caso seja necessário, será solicitado auxílio da diretoria da escola.

Procedimentos: Será realizada uma oficina de Introdução à Ciência da Computação no laboratório de informática da escola. **Serão realizados 4-6 encontros, com a duração de uma hora cada.** As atividades realizadas pelos participantes serão acompanhadas pelo pesquisador, que fará observações, anotações e questionamentos. **Dessas atividades, a principal atividade é jogar um jogo eletrônico com temática relacionada ao pensamento computacional.** A princípio, os encontros da oficina serão filmados; isso possibilitará uma melhor compreensão e interpretação das atividades e, conseqüentemente, uma melhor oportunidade de sucesso para a pesquisa como um todo. **Os participantes serão informados e familiarizados sobre as filmagens. Caso eles não se sintam à vontade por conta das gravações, as filmagens não ocorrerão.**

Alternativas:

- A autorização para participar da pesquisa é voluntária. A recusa em autorizar não acarretará em quaisquer penalidades, mas impedirá a participação na pesquisa;
- Os dados pessoais não serão identificados em nenhuma publicação que resultar deste estudo;

- A efetivação do envolvimento com esta pesquisa somente se dará a partir da assinatura deste termo, com o qual estará consentida a participação do aluno, sendo-lhe reservado o direito de recusar-se a participar ou de desistir de sua participação a qualquer momento;
- A desistência ou não participação do aluno não lhe causará nenhum prejuízo; os dados da pesquisa referentes aos alunos desistentes serão descartados.

Confidencialidade: O pesquisador e sua orientadora certificam que todos os dados desta pesquisa serão mantidos em sigilo. Eventualmente, os resultados desta pesquisa podem ser publicados em eventos ou periódicos científicos; mesmo nesses casos, a confidencialidade será preservada. De acordo com a Resolução 466/12, fica garantido o sigilo absoluto das informações obtidas nessa pesquisa, sendo a guarda dos materiais produzidos de inteira responsabilidade do pesquisador – que se encarregará de armazená-los sob proteção criptográfica. Após o período de cinco anos, as anotações, as filmagens e quaisquer registros produzidos pelos participantes serão descartados.

Problemas e perguntas: O pesquisador, Luís Filipe Severgnini, é bacharel em Tecnologias Digitais, técnico em Informática e mestrando do programa de Pós-graduação em Educação da Universidade de Caxias do Sul, sob orientação da professora Dra. Eliana Maria do Sacramento Soares, e se compromete a esclarecer devida e adequadamente qualquer dúvida ou necessidade de informações que o participante ou responsável venham a ter no momento da pesquisa ou posteriormente através do telefone (54) 99204-1975 ou pelo endereço de e-mail lfsevergnini@ucs.br. Também é possível contatar o Comitê de Ética em Pesquisa pelo telefone **(54) 3218-2829**.

Custos: A participação na pesquisa não acarretará em custos, assim como não será disponibilizada nenhuma forma de compensação financeira.

Termo de Consentimento

Após ter sido devidamente informado(a) de todos os aspectos da pesquisa e ter esclarecido todas as minhas dúvidas, concordo em autorizar a participação da criança pela qual sou responsável na referida pesquisa e a participar das atividades propostas.

Nome legível do responsável pelo estudante:

Assinatura do responsável:

Atesto que expliquei a natureza e o objetivo de tal estudo, bem como os possíveis riscos e benefícios do mesmo, junto ao participante e seu respectivo responsável. Penso que todas as informações necessárias lhes foram fornecidas em uma linguagem adequada e compreensível, e que ambos compreenderam do que a pesquisa se trata.

Endereços para contato:

E-mail: lfsevergnini@ucs.br

Fone: (54) 99204-1975

Pesquisador responsável:

Nome legível: Luís Filipe Severgnini

Assinatura: _____

Este projeto está vinculado ao Comitê de Ética em Pesquisa da Universidade de Caxias do Sul, sob o endereço Rua Francisco Getúlio Vargas, 1130, Bloco M, sala 106, Caxias do Sul, RS, CEP: 95070-560, fone (54) 3218-2829.

APÊNDICE D – Termo de Assentimento Livre e Esclarecido

APÊNDICE C – TERMO DE ASSENTIMENTO LIVRE E ESCLARECIDO

O termo de assentimento livre e esclarecido será entregue a cada um dos participantes da oficina, os quais deverão assinar e rubricar cada uma das vias. Uma das vias será retida pelo pesquisador.

Projeto de pesquisa: jogos eletrônicos (*serious games*) como elemento mediador do desenvolvimento do pensamento computacional.

Pesquisador responsável: Luís Filipe Severgnini

Orientadora: Dra. Eliana Maria do Sacramento Soares

Instituição: Universidade de Caxias do Sul

Objetivo: Identificar as contribuições dos jogos eletrônicos (*serious games*) no desenvolvimento do pensamento computacional e, a partir disso, elaborar uma proposta de prática pedagógica em que jogos desse gênero participem como elementos mediadores do desenvolvimento de tal forma de pensamento.

Benefícios: O principal benefício desta pesquisa é produzir novos conhecimentos relacionados ao desenvolvimento do pensamento computacional, uma abordagem para solução de problemas que envolve o uso de um conjunto de práticas e princípios da ciência da computação para formular soluções executáveis por um computador, não exclusiva para programadores, e aplicável em uma variedade diversificada de campos. Compreender melhor esse assunto pode contribuir para que um número maior de estudantes sejam criadores de tecnologias em vez de apenas usuários. Ao mesmo tempo, serão investigadas as contribuições dos jogos eletrônicos (*serious games*) nesse processo, o que pode abrir espaço para uma forma mais lúdica, prazerosa e motivadora de aprender o pensamento computacional.

Riscos: Entende-se que os riscos da pesquisa sejam mínimos, pois ela não aborda nenhum assunto moralmente controverso e tampouco prevê atividades físicas que possam causar lesões e/ou traumas. No entanto, eventualmente, as atividades podem gerar algum desconforto, *stress*, cansaço ou tédio nos participantes. Por isso, estes podem desistir de participar das atividades – ou até mesmo da pesquisa como um todo – a qualquer momento, sem nenhum prejuízo de sua parte. A seguir, descreve-se com detalhamento os riscos envolvidos na pesquisa:

- **DESCONFORTO, STRESS OU CANSAÇO.** Prevenção: esclarecer que as atividades da oficina não têm caráter avaliativo e não estão vinculadas às atividades regulares e, por isso, não há necessidade para nervosismo ou medo de errar; motivar os sujeitos a buscar a diversão em primeiro lugar. Reparação: aumentar o tempo de cada atividade; incentivar os sujeitos a trabalharem em conjunto; acompanhar a realização da atividade do sujeito; interromper a realização da atividade.
- **TÉDIO.** Prevenção: acompanhar os movimentos dos sujeitos com o intuito de perceber frustrações com o jogo, seja por causa do nível de dificuldade ou por desinteresse no jogo em si. Reparação: em caso de algum desafio se tornar frustrante, causando ansiedade nos sujeitos, fornecer pistas e realizar problematizações de modo a possibilitar que o desafio seja superado. Caso os jogadores achem o jogo desinteressante, existe a possibilidade de substituir o jogo.
- **FRUSTRAÇÃO EM NÃO CONSEGUIR CUMPRIR AS ATIVIDADES PROPOSTAS.** Prevenção: esclarecer as atividades propostas; estabelecer um ambiente acolhedor que incentive a tentativa e o erro por parte dos alunos, bem como a elaboração de perguntas quando houver dúvidas ou curiosidades. Reparação: explicar que não há problemas em errar, enfatizando que o erro faz parte da aprendizagem; caso o sujeito esteja disposto a continuar, incentivar nova tentativa; em caso de repetidas frustrações, adequar a atividade para que atenda o nível de desenvolvimento do sujeito.

- **CONSTRANGIMENTOS DE EXPOSIÇÃO.** Prevenção: esclarecer aos sujeitos que ninguém além do pesquisador terá acesso às gravações da oficina; esclarecer que as gravações estarão armazenadas de maneira segura; em caso de ausência de acordo mútuo dos sujeitos, suspender as gravações. Reparação: se os sujeitos demonstrarem desconforto ou insatisfação com a exposição gerada pelas filmagens, estas serão suspensas.
- **CONFLITO ENTRE OS PARTICIPANTES.** Prevenção: incentivar a cooperação entre os participantes e, em caso de existir competições ou rivalidades no grupo, encorajar que a competição seja salutar. Reparação: retomar as ações preventivas. Caso seja necessário, será solicitado auxílio da diretoria da escola.

Procedimentos: Será realizada uma oficina de Introdução à Ciência da Computação no laboratório de informática da escola. **Serão realizados 4-6 encontros, com a duração de uma hora cada.** As atividades realizadas pelos participantes serão acompanhadas pelo pesquisador, que fará observações, anotações e questionamentos. **Dessas atividades, a principal atividade é jogar um jogo eletrônico com temática relacionada ao pensamento computacional.** A princípio, os encontros da oficina serão filmados; isso possibilitará uma melhor compreensão e interpretação das atividades e, conseqüentemente, uma melhor oportunidade de sucesso para a pesquisa como um todo. **Os participantes serão informados e familiarizados sobre as filmagens. Caso eles não se sintam à vontade por conta das gravações, as filmagens não ocorrerão.**

Alternativas:

- A autorização para participar da pesquisa é voluntária. A recusa em autorizar não acarretará em quaisquer penalidades, mas impedirá a participação na pesquisa;
- Os dados pessoais não serão identificados em nenhuma publicação que resultar deste estudo;

- A efetivação do envolvimento com esta pesquisa somente se dará a partir da assinatura deste termo, com o qual estará consentida a participação do aluno, sendo-lhe reservado o direito de recusar-se a participar ou de desistir de sua participação a qualquer momento;
- A desistência ou não participação do aluno não lhe causará nenhum prejuízo; os dados da pesquisa referentes aos alunos desistentes serão descartados.

Confidencialidade: O pesquisador e sua orientadora certificam que todos os dados desta pesquisa serão mantidos em sigilo. Eventualmente, os resultados desta pesquisa podem ser publicados em eventos ou periódicos científicos; mesmo nesses casos, a confidencialidade será preservada. De acordo com a Resolução 466/12, fica garantido o sigilo absoluto das informações obtidas nessa pesquisa, sendo a guarda dos materiais produzidos de inteira responsabilidade do pesquisador – que se encarregará de armazená-los sob proteção criptográfica. Após o período de cinco anos, as anotações, as filmagens e quaisquer registros produzidos pelos participantes serão descartados.

Problemas e perguntas: O pesquisador, Luís Filipe Severgnini, é bacharel em Tecnologias Digitais, técnico em Informática e mestrando do programa de Pós-graduação em Educação da Universidade de Caxias do Sul, sob orientação da professora Dra. Eliana Maria do Sacramento Soares, e se compromete a esclarecer devida e adequadamente qualquer dúvida ou necessidade de informações que o participante ou responsável venham a ter no momento da pesquisa ou posteriormente através do telefone (54) 99204-1975 ou pelo endereço de e-mail lfsevergnini@ucs.br. Também é possível contatar o Comitê de Ética em Pesquisa pelo telefone **(54) 3218-2829**.

Custos: A participação na pesquisa não acarretará em custos, assim como não será disponibilizada nenhuma forma de compensação financeira.

Termo de Assentimento

Após ter sido devidamente informado(a) de todos os aspectos da pesquisa e ter esclarecido todas as minhas dúvidas, concordo em participar da referida pesquisa e participar das atividades propostas.

Estarei assinando as duas vias, sendo uma retida por mim, e a outra pelo pesquisador.

Eu, _____ declaro meu consentimento em participar desta pesquisa.

Assinatura do participante: _____

Atesto que expliquei a natureza e o objetivo de tal estudo, bem como os possíveis riscos e benefícios do mesmo, junto ao participante e seu respectivo responsável. Penso que todas as informações necessárias lhes foram fornecidas em uma linguagem adequada e compreensível, e que ambos compreenderam do que a pesquisa se trata.

Endereços para contato:

E-mail: lfsevergnini@ucs.br

Fone: (54) 99204-1975

Pesquisador responsável:

Nome legível: Luís Filipe Severgnini

Assinatura: _____

APÊNDICE E – Programa em Python nº1

```

import csv
import pprint
from prettytable import PrettyTable

alunos = {}
conceitos = {}

# Lê os níveis resolvidos por cada aluno
with open('niveis_alunos.csv') as arquivo:
    reader = csv.DictReader(arquivo, delimiter=';')
    for linha in reader:
        # Colunas depois de Nivel e Titulo representam alunos
        if not alunos:
            for aluno in [col for col in linha if col != 'Nivel' and col
                ↪ != 'Titulo']:
                alunos[aluno] = { 'niveis': [] }

        for aluno in alunos:
            if linha[aluno] != '':
                alunos[aluno]['niveis'].append(linha['Nivel'])

# Lê os conceitos trabalhados em cada nível
with open('niveis_conceitos_pc.csv') as arquivo:
    reader = csv.DictReader(arquivo, delimiter=';')
    for linha in reader:
        # Colunas depois de Nivel e Titulo representam conceitos
        if not conceitos:
            for conceito in [col for col in linha if col != 'Nivel' and
                ↪ col != 'Titulo']:
                conceitos[conceito] = { 'niveis': [] }

        for conceito in conceitos:
            if linha[conceito] != '':
                conceitos[conceito]['niveis'].append(linha['Nivel'])

```

```

# Confere quais conceitos foram trabalhados por quais alunos
for (descricao, conceito) in conceitos.items():
    maxMarcas = len(conceito['niveis'])
    # print(descricao)

    for (nome, aluno) in alunos.items():
        marcas =
        ↪ len(set(conceito['niveis']).intersection(aluno['niveis']))
        aproveitamento = round(marcas / maxMarcas * 100, 2)

        # Salva o aproveitamento do aluno neste conceito
        if 'conceitos' not in aluno:
            aluno['conceitos'] = {}

        aluno['conceitos'][descricao] = { 'marcas': marcas,
        ↪ 'aproveitamento': aproveitamento }
        # print(' * ', nome, ': ', marcas, '/', maxMarcas, ' = ',
        ↪ aproveitamento, '%', sep='')

# -----
# Exibe o cruzamento dos dados
# -----
t = PrettyTable(['Conceito', 'Marcas'] + list(alunos.keys()));

for (descricao, conceito) in conceitos.items():
    t.add_row([descricao, len(conceito['niveis'])] +
    ↪ [aluno['conceitos'][descricao]['aproveitamento'] for aluno in
    ↪ alunos.values()])

# Totais
t.add_row(['Totais', ''] + [round(sum([conceito['aproveitamento'] for
    ↪ conceito in aluno['conceitos'].values()]) / len(conceitos), 2) for
    ↪ aluno in alunos.values()])

# Imprime a tabela
print(t)

# -----
# Debugging

```

```
# -----  
# pp = pprint.PrettyPrinter()  
# pp.pprint(alunos)  
# pp.pprint(conceitos)
```

APÊNDICE F – niveis_alunos.csv

```

Nivel;Titulo;Ermac;Colega/Ichigo;Johan/Honda;XVds/Bags;Knight;Donalds;Cheetos
1;Dungeons of Kithgard;X;X;X;X;X;X;X;X
2;Gems in the Deep;X;X;X;X;X;X;X;X
3;Shadow Guard;X;X;X;X;X;X;X;X
3a;Kounter Kithwise;;X;;;X;X;
3b;Crawlways of Kithgard;;X;;;X;X;
Conceito A;Careful Steps;X;;X;X;X;X;X
4;Enemy Mine;X;X;X;X;X;X;X;X
4a;Illusory Interruption;;;X;X;X;
4b;Forgetful Gemsmith;;;X;X;
Conceito B;Long Steps;X;;X;X;X;X;
5;True Names;X;X;X;X;X;X;X;X
5a;Favorable Odds;X;X;X;X;X;X;X;X
5b;The Raised Sword;;X;;;X;X;
Conceito C;Dangerous Steps;;;X;X;X;X;X
Combo A;Sleep Hour;X;;X;;X;X;
6;Cell Commentary;X;X;X;X;X;X;X;X
7;Kithgard Librarian;X;X;X;X;X;X;X;X
8;The Prisoner;X;X;X;X;X;X;X;X
9;Fire Dancing;X;X;X;X;X;X;X;X
10;Haunted Kithmaze;X;X;X;X;X;X;X;X
11;Descending Further;X;X;X;X;X;X;X;X
11a;Riddling Kithmaze;X;X;X;X;X;X;X;X
11b;Radiant Aura;;X;X;X;X;X;
Conceito D;Loop Warehouse;X;;X;X;X;X;X
12;Dread Door;X;X;X;X;X;X;X;X
13;Hack and Dash;X;X;X;X;X;X;X;X
14;Cupboards of Kithgard;X;X;X;X;X;X;X;X
14a;Cupboards of Kithgard A;;X;X;X;X;X;X;
14b;Cupboards of Kithgard B;;X;;X;X;X;X;
15;Known Enemy;X;X;X;X;X;X;X;X
16;Master of Names;X;X;X;X;X;X;X;X
16a;Lowly Kithmen;X;X;X;X;X;X;X;X
16b;Closing the Distance;X;X;;X;;X;
Conceito E;Master of Names Debug;;;X;;X;X;

```

17;The Final Kithmaze;X;X;X;X;X;X;
Combo B;Safe Place;X;;X;;X;X;
18;Kithgard Gates;X;X;X;X;X;X;
Arena;Wakka Maul;X;X;X;X;X;;

APÊNDICE G – níveis_conceitos_pc.csv

```

Nivel;Titulo;Pens. Logico;Pens. Algo.;Decomposicao;Rec. Padroes e
↳ General.;Abstracao e Model.;Avaliacao
1;Dungeons of Kithgard;;;;X;
2;Gems in the Deep;;X;X;;
3;Shadow Guard;X;X;;;
3a;Kounter Kithwise;;;;X;;
3b;Crawlways of Kithgard;;;;X;;
Conceito A;Careful Steps;;;;X;;
4;Enemy Mine;;;;;X
4a;Illusory Interruption;;;;X;;
4b;Forgetful Gemsmith;;;;X;;
Conceito B;Long Steps;;X;X;X
5;True Names;X;;;X;X;
5a;Favorable Odds;;;;X;;
5b;The Raised Sword;;;;X;;
Conceito C;Dangerous Steps;;;;X;;
Combo A;Sleep Hour;;X;X;X;X
6;Cell Commentary;;;;;
7;Kithgard Librarian;;;;X;;
8;The Prisoner;X;X;X;X;;
9;Fire Dancing;X;;;;;X
10;Haunted Kithmaze;;;;X;X
11;Descending Further;;;;X;;
11a;Riddling Kithmaze;;;;X;;
11b;Radiant Aura;;;;X;;
Conceito D;Loop Warehouse;;X;X;X;X
12;Dread Door;;;;X;;
13;Hack and Dash;;;X;X;;
14;Cupboards of Kithgard;;;;;X;
14a;Cupboards of Kithgard A;;;;X;;
14b;Cupboards of Kithgard B;;;;X;;
15;Known Enemy;;;;;X;
16;Master of Names;;;;;X;
16a;Lowly Kithmen;;;;X;;
16b;Closing the Distance;;;;X;;

```


Conceito E;Master of Names Debug;;;X;;X
17;The Final Kithmaze;X;;;X;;
Combo B;Safe Place;;;X;X;
18;Kithgard Gates;;X;X;;;
Arena;Wakka Maul;X;X;X;X;X;X

APÊNDICE H – Programa em Python nº2

```

import csv
import pprint
from prettytable import PrettyTable

pesos = {}
alunos = {}
conceitos = {}

# Lê os pesos dos pareceres
with open('legendas_quali_avaliacao.csv') as arquivo:
    reader = csv.DictReader(arquivo, delimiter=';')
    for linha in reader:
        pesos[linha['Parecer']] = { 'peso': float(linha['Peso']),
        ↪ 'legenda': linha['Legenda'] }

# Lê os níveis resolvidos por cada aluno
with open('niveis_alunos_quali.csv') as arquivo:
    reader = csv.DictReader(arquivo, delimiter=';')
    for linha in reader:
        # Colunas depois de Nivel e Titulo representam alunos
        if not alunos:
            for aluno in [col for col in linha if col != 'Nivel' and col
            ↪ != 'Titulo' and col != 'Peso']:
                alunos[aluno] = { 'niveis': [] }

        for aluno in alunos:
            if linha[aluno] in pesos:
                alunos[aluno]['niveis'].append({ 'nivel': linha['Nivel'],
                ↪ 'nota': float(pesos[linha[aluno]]['peso']) *
                ↪ float(linha['Peso'])})

# Lê os conceitos trabalhados em cada nível
with open('niveis_conceitos_pc_quali.csv') as arquivo:
    reader = csv.DictReader(arquivo, delimiter=';')
    for linha in reader:

```

```

# Colunas depois de Nível e Título representam conceitos
if not conceitos:
    for conceito in [col for col in linha if col != 'Nível' and
↪ col != 'Título' and col != 'Peso']:
        conceitos[conceito] = { 'niveis': [] }

for conceito in conceitos:
    if linha[conceito] != '':
        conceitos[conceito]['niveis'].append({ 'nivel':
↪ linha['Nível'], 'peso': float(linha['Peso']) })

# Confere quais conceitos foram trabalhados por quais alunos
for (descricao, conceito) in conceitos.items():
    conceito['nota_maxima'] = sum([nivel['peso'] for nivel in
↪ conceito['niveis']])

for (nome, aluno) in alunos.items():
    marcas = sum(nivel['nota'] for nivel in aluno['niveis'] if
↪ nivel['nivel'] in [n['nivel'] for n in conceito['niveis']])
    aproveitamento = round(marcas / conceito['nota_maxima'] * 100, 2)

# Salva o aproveitamento do aluno neste conceito
if 'conceitos' not in aluno:
    aluno['conceitos'] = {}

aluno['conceitos'][descricao] = { 'marcas': marcas,
↪ 'aproveitamento': min(aproveitamento, 100.0) }
# print(' * ', nome, ': ', marcas, '/', maxMarcas, ' = ',
↪ aproveitamento, '%', sep='')

# -----
# Exibe o cruzamento dos dados
# -----
t = PrettyTable(['Conceito', 'Marcas'] + list(alunos.keys()));

for (descricao, conceito) in conceitos.items():
    t.add_row([descricao, conceito['nota_maxima']] +
↪ [aluno['conceitos'][descricao]['aproveitamento'] for aluno in
↪ alunos.values()])

```

```
# Totais
t.add_row(['Totais', sum(conceito['nota_maxima'] for conceito in
→ conceitos.values())] + [round(sum([conceito['aproveitamento'] for
→ conceito in aluno['conceitos'].values()) / len(conceitos), 2) for
→ aluno in alunos.values()]

# Imprime a tabela
print(t)

# -----
# Debugging
# -----
# pp = pprint.PrettyPrinter()
# pp.pprint(pesos)
# pp.pprint(alunos)
# pp.pprint(conceitos)
```

APÊNDICE I

– niveis_conceitos_pc_quali.csv

```

Nivel;Titulo;Peso;Pens. Logico;Pens. Algo.;Decomposicao;Rec. Padroes e
↔ General.;Abstracao e Model.;Avaliacao
1;Dungeons of Kithgard;1;;;;;X;
2;Gems in the Deep;1;;X;;X;;
3;Shadow Guard;1;X;X;;;;;
3a;Kounter Kithwise;0.5;;;;;X;;
3b;Crawlways of Kithgard;0.5;;;;;X;;
Conceito A;Careful Steps;2;;;;;X;;
4;Enemy Mine;1;;;;;;X
4a;Illusory Interruption;0.5;;;;;X;;
4b;Forgetful Gemsmith;0.5;;;;;X;;
Conceito B;Long Steps;2;;X;;X;;X
5;True Names;1;X;;;X;X;
5a;Favorable Odds;0.5;;;;;X;;
5b;The Raised Sword;0.5;;;;;X;;
Conceito C;Dangerous Steps;2;;;;;X;;
Combo A;Sleep Hour;3;;X;X;X;;X
6;Cell Commentary;1;;;;;;
7;Kithgard Librarian;1;;;;;X;;
8;The Prisoner;1;X;X;X;X;;
9;Fire Dancing;1;X;;;;;X
10;Haunted Kithmaze;1;;;;;X;;X
11;Descending Further;1;;;;;X;;
11a;Riddling Kithmaze;0.5;;;;;X;;
11b;Radiant Aura;0.5;;;;;X;;
Conceito D;Loop Warehouse;2;;X;X;X;;X
12;Dread Door;1;;;;;X;;
13;Hack and Dash;1;;;X;X;;
14;Cupboards of Kithgard;1;;;;;X;
14a;Cupboards of Kithgard A;0.5;;;;;X;;
14b;Cupboards of Kithgard B;0.5;;;;;X;;
15;Known Enemy;1;;;;;X;
16;Master of Names;1;;;;;X;

```

16a;Lowly Kithmen;0.5;;;X;;
16b;Closing the Distance;0.5;;;X;;
Conceito E;Master of Names Debug;2;;;X;;X
17;The Final Kithmaze;1;X;;;X;;
Combo B;Safe Place;3;;;X;X;
18;Kithgard Gates;1;;X;X;;;
Arena;Wakka Maul;3;X;X;X;X;X;X

APÊNDICE J – niveis_alunos_quali.csv

Nivel;Titulo;Peso;Ermac;Colega/Ichigo;Johan/Honda;XVds/Bags;Knight;Donalds;Cheetos
 1;Dungeons of Kithgard;1;OK;OK;OK;OK;OK;OK;OK
 2;Gems in the Deep;1;NO;OK;OK;OK;NO;NO;OK
 3;Shadow Guard;1;OK;OK;OK;OK;OK;OK;OK
 3a;Kounter Kithwise;0.5;-;OK;-;-;OK;OK;-
 3b;Crawlways of Kithgard;0.5;-;NO;-;-;OK;OK;-
 Conceito A;Careful Steps;2;OK *;-;OK *;OK *;OK *;OK MKO
 4;Enemy Mine;1;OK;OK;OK;OK;OK;OK;OK
 4a;Illusory Interruption;0.5;-;-;-;OK;OK;OK;-
 4b;Forgetful Gemsmith;0.5;-;-;-;-;OK;OK;-
 Conceito B;Long Steps;2;OK;-;OK;OK;OK;OK;-
 5;True Names;1;OK;OK;OK;OK;OK;OK;OK
 5a;Favorable Odds;0.5;OK;OK +;OK;OK +;OK +;OK +;OK
 5b;The Raised Sword;0.5;-;OK;-;-;OK;OK;-
 Conceito C;Dangerous Steps;2;-;-;NO;OK;NO;NO;OK
 Combo A;Sleep Hour;3;OK *;-;OK *;-;OK;OK +;-
 6;Cell Commentary;1;OK;OK +;OK;OK Cr;OK +;OK +;OK +
 7;Kithgard Librarian;1;OK;OK;OK;OK Cr;OK;OK +;OK +
 8;The Prisoner;1;NO *;OK;OK +;NO *;OK;OK;OK MKO
 9;Fire Dancing;1;OK;OK;OK;OK;OK;NO;OK
 10;Haunted Kithmaze;1;OK;OK +;OK;OK +;OK +;NO;OK +
 11;Descending Further;1;OK;OK;OK;OK;OK;OK;OK
 11a;Riddling Kithmaze;0.5;OK;OK;OK;OK;OK;OK;OK
 11b;Radiant Aura;0.5;-;OK;OK;OK;NO Cr;OK;-
 Conceito D;Loop Warehouse;2;OK MKO;-;OK MKO;OK;OK;OK;-
 12;Dread Door;1;OK;OK;OK;OK;OK;OK;OK
 13;Hack and Dash;1;OK MKO;OK MKO;OK MKO;OK;OK MKO;OK;NC
 14;Cupboards of Kithgard;1;OK;NO;OK;OK;OK;OK;-
 14a;Cupboards of Kithgard A;0.5;-;OK;OK;OK;OK;OK;-
 14b;Cupboards of Kithgard B;0.5;-;OK;-;OK;OK;OK;-
 15;Known Enemy;1;OK;OK *;OK;OK;OK;OK;-
 16;Master of Names;1;OK +;OK +;OK;OK *;OK *;OK;-
 16a;Lowly Kithmen;0.5;OK *;OK *;OK *;OK *;OK *;OK Cr;-
 16b;Closing the Distance;0.5;NO;NO;-;NO;-;OK;-
 Conceito E;Master of Names Debug;2;-;-;OK;-;OK *;OK;-

17;The Final Kithmaze;1;OK;OK;OK;OK MKO;OK Cr;OK;-
Combo B;Safe Place;3;NC;-;OK -;-;OK +;NC;-
18;Kithgard Gates;1;OK;OK *;OK Cr;OK Cr;OK;OK;-
Arena;Wakka Maul;3;OK;OK;OK;OK;OK;OK -;-

APÊNDICE K

– legendas_quali_avaliacao.csv

Peso;Parecer;Legenda

1.1;OK +;Solucao correta, melhor otimizada

1.1;OK Cr;Solucao correta e criativa

1.05;OK *;Solucao correta, caminho alternativo

1;OK;Solucao corresponde a esperada

0.95;NO *;Solucao correta, nao otimizada, caminho alternativo

0.9;NO;Solucao correta, mas nao otimizada

0.8;OK MKO;Solucao correta, feita em conjunto com um MKO

0.75;OK -;Solucao correta, mas incompleta

0.25;NC;Nao concluido

0;-;Nao realizado