

Uso de Jogos Sérios no Ensino de Lógica de Programação: uma revisão sistemática da literatura

Marlow R. B. Dickel¹, Avanilde Kemczinski², Marcelo da Silva Hounsell²

¹Centro Universitário SENAI/SC – Campus Joinville

² PPGCAP – Programa de Pós-Graduação em Computação Aplicada – UDESC –
Universidade do Estado de Santa Catarina

profmarlow@gmail.com, {avanilde.kemczinski,marcelo.hounsell}@udesc.br

Resumo. *Este artigo descreve uma revisão sistemática da literatura sobre o uso de jogos sérios no ensino de lógica de programação. Dos 402 artigos encontrados sobre o tema, foram selecionados quatro publicados a partir do ano de 2013. Havia poucos jogos para analisar, mas nenhum gênero ou plataforma de jogo preferido foi identificado. No entanto, todos os jogos se concentraram nos fundamentos da lógica de programação, embora tenham sido avaliados em uma ampla variedade de métricas. Conclui-se que, apesar de possíveis ameaças à pesquisa, o desenvolvimento de jogos para auxiliar no ensino de lógica de programação é uma área que ainda está incipiente e lida com conceitos muito introdutórios.*

Palavras-chave— *Jogos sérios, Lógica de programação, Ensino, Revisão Sistemática.*

Abstract. *This paper describes a systematic literature review on the use of serious games in teaching programming logic. Out of 402 papers found, four published from the year 2013 were selected. There were just a few games to analyze but no preferred game genre or platform was identified. However, all games focused on the very basics of programming logic, although they have been assessed in a wide variety of metrics. It is concluded that, despite possible threats to the research, the development of games to help teach programming logic is an area that is in its infancy and dealing with introductory concepts.*

Keywords— *Serious games, Programming logic, Teaching, Systematic Review.*

1. Introdução

Uma das mais importantes habilidades atualmente nas áreas tecnológicas é a de conhecer linguagens de programação, visando conseguir desenvolver programas para computador, dispositivos móveis, ou algum outro equipamento eletrônico. Araújo e Schwamborn (2013) afirmam que o desenvolvimento dessas habilidades de programação “[...] nos dias atuais é tão importante quanto aprender a ler e a escrever”, e se inicia através da lógica de programação, uma das matérias iniciais em qualquer curso da área de desenvolvimento de *software*. O bom desenvolvimento do estudante nesta etapa possibilita a ele o ingresso posterior nos estudos de qualquer linguagem de programação com maior facilidade.

O uso de jogos na educação vem se mostrando um meio de motivar e potencializar o ensino. Jogos sérios, uma categoria de jogos cuja intenção é atingir um objetivo além do entretenimento, vem sendo bastante utilizados na área da educação com efeitos

positivos, devido ao foco maior no resultado educacional e no conteúdo a ser lecionado, e atualmente são vistos como complemento às tradicionais formas de ensino (MASKELIŪNAS et al., 2020).

Este trabalho tem como objetivo realizar um levantamento bibliográfico sobre o **uso de jogos sérios no ensino da lógica de programação**. Para isso, foi feita uma revisão sistemática dos artigos que tratam o tema de 2013 até o presente. Três perguntas secundárias serão respondidas: Que tipos de jogos sérios são utilizados atualmente no ensino de lógica de programação? Quais as características dos jogos sérios utilizados atualmente no ensino de lógica de programação? e Quais as métricas empregadas para avaliar o uso de jogos sérios no ensino de lógica de programação?

Dos trabalhos relacionados ao tema, de Oliveira, Stringhini e Corrêa (2018) realizaram uma revisão sistemática da literatura brasileira sobre estratégias do ensino de lógica de programação, e Borges et al. (2018) buscaram entender as metodologias, práticas e ferramentas no ensino de lógica de programação usadas no Brasil também através de uma revisão sistemática, porém ambos não observaram apenas trabalhos sobre jogos sérios e não consultaram a literatura estrangeira, pontos importantes desta pesquisa.

1.1 Metodologia

Uma revisão sistemática sobre o tema proposto foi realizada no período de abril a junho de 2022. O protocolo da revisão foi baseado em Klock (2018). As palavras-chave do argumento de busca foram definidas com base no objetivo do trabalho. A frase de busca resultante foi: (*gam* AND "Programming logic" AND ("Teaching" OR "Learning")*).

Os mecanismos de busca utilizados foram selecionados com base em Buchinger, Cavalcanti e Hounsell (2014), que realizaram uma análise quantitativa dos recursos disponíveis em 40 mecanismos. Dos sete mecanismos melhor avaliados pelos autores, apenas o Web of Knowledge não foi utilizado devido ao seu acesso restrito. Assim, seis mecanismos de busca foram utilizados: Engineering Village, Scopus SciVerse, IEEE Xplore, ACM Digital Library, Science Direct e Springer Link.

Os critérios de seleção para filtragem dos estudos foram divididos em critérios de inclusão e exclusão, e são apresentados na Tabela 1, a seguir:

Tabela 1. Critérios de inclusão e exclusão de estudos

Tipo	ID	Descrição
Inclusão	CI1	Artigos escritos em inglês ou português
	CI2	Artigos primários
	CI3	Artigos completos (<i>full paper</i>)
	CI4	Artigos publicados em revistas ou conferências
Exclusão	CE1	Artigos que não aplicam jogos sérios no ensino de lógica de programação
	CE2	Artigos não disponíveis para acesso na íntegra
	CE3	Artigos duplicados
	CE4	Artigo publicado antes do ano de 2013

Os dados a serem extraídos de cada trabalho selecionado terão como objetivo responder as três perguntas descritas como objetivo do trabalho, sendo, portanto: os tipos, características dos jogos sérios utilizados atualmente no ensino de lógica de programação, bem como as métricas empregadas para avaliar seu uso.

2. Aplicação da metodologia

Primeiramente, foi realizada a busca nas ferramentas selecionadas por artigos que atendessem o tema proposto, através da frase de busca definida anteriormente. Uma exceção ocorreu na plataforma Science Direct, onde a frase foi alterada para ("*serious games*" OR "*serious game*") AND "*Programming logic*" AND ("*Teaching*" OR "*Learning*") pois o caractere coringa '*' não é aceito.

Em seguida, foram aplicados os critérios de inclusão e exclusão. Nessa filtragem, foram identificados apenas quatro artigos que se aplicam nos critérios de inclusão e exclusão. A Tabela 2 exibe os resultados obtidos pelas 6 ferramentas, a atualização dos resultados após aplicar o filtro por ano de publicação e, por fim, os trabalhos considerados relevantes após a aplicação dos demais critérios de inclusão e exclusão.

Tabela 2. Resultado quantitativo das buscas nas plataformas selecionadas

Plataforma	Resultados encontrados	Publicações de 2013 a 2022	Trabalhos relevantes
Engineering Village	14	13	3
Scopus SciVerse	14	13	1
IEEE Xplore	10	4	0
ACM Digital Library	161	90	0
Science Direct	19	18	0
Springer Link	184	105	0
Total	402	243	4

A Tabela 3, apresenta o objetivo, resultados, e a origem dos trabalhos relevantes, com base em seus resumos.

Tabela 3. Objetivo e resultados dos trabalhos relevantes

Autoria	Objetivo	Resultados	Origem
Silva et al. (2020)	Propor um jogo educacional aberto, para estudantes que nunca tiveram contato ou apresentam alguma dificuldade em lógica de programação	Estudantes tiveram boas experiências durante o jogo, que teve impacto positivo em seu aprendizado	Brasil
Tsai e Lai (2022)	Propor um sistema de ensino de lógica de programação em realidade aumentada (RA) que combina tecnologia RA e designs de material de ensino baseado em jogos com base nos conceitos fundamentais de programação estruturada para a sétima série	Os resultados do experimento validam que o sistema de ensino proposto teve efeitos positivos significantes em potencializar a efetividade e motivação do aprendizado dos estudantes	Taiwan
Lee e Quek (2018)	Propor o uso de Realidade Aumentada Tangível (RAT) para criar um jogo interativo chamado TARogic que ensina aos estudantes o básico em lógica de programação	60% dos estudantes foram positivos quanto a sua experiência geral de aprendizado usando o TARogic	Malásia
Hong e Chu (2017)	Desenvolver uma abordagem em Game-Based Learning (GBL) para solução de problemas computacionais e programação situada em 3D para envolver os alunos no aprendizado e na prática de suas habilidades de programação.	Concluem que o uso da abordagem sugerida pode efetivamente diminuir a carga cognitiva dos alunos e envolvê-los no jogo para concluir o aprendizado da lógica de programação no curso	Taiwan

3. Resultados

Para responder as perguntas secundárias desta pesquisa, referindo-se aos tipos, características e métricas de avaliação, ao invés de se usar uma métrica consolidada para comparar os trabalhos, são usados os pontos de vista dos autores dos trabalhos relevantes, possibilitando assim também discutir posteriormente os pontos de vista de cada autor sobre como definir os tipos, as características e métricas de avaliação de um jogo sério no ensino de lógica de programação.

Respondendo a pergunta secundária “Que tipos de jogos sérios são utilizados atualmente no ensino de lógica de programação?”, o jogo apresentado no artigo de Silva et al. (2020) é um jogo do tipo RPG (*Role Playing Game*) para computador, desenvolvido na *engine* Unity com a linguagem de programação C# e com seus personagens e cenários criados com o programa Tiled; no artigo de Tsai e Lai (2022) é apresentado um jogo em Realidade Aumentada (RA) desenvolvido para Android com ARCore, onde o estudante deve conduzir um objeto virtual até o final das fases, mas o tipo do jogo não é identificado pelos autores; o artigo de Lee e Quek (2018) demonstra o uso de um jogo de RA cujo objetivo também é mover um elemento do início ao fim das fases, e também não há caracterização do tipo do jogo pelos autores, mas é explicado que sua construção foi dividida em 2 partes: console (composto por uma placa Arduino Uno com botões e módulos USB de leitura e escrita e Bluetooth, *pendrives* para realizar as entradas de comandos no jogo, e um *Head-Mounted Display* (HMD) com *smartphone* para a visualização do jogo em RA) e jogo (desenvolvido em Unity3D para mapas, personagens e mecânica do jogo, com o plugin Vuforia para criação e detecção dos marcadores AR); e o artigo de Hong e Chu (2017), onde um jogo de categoria *puzzle* é apresentado, não se menciona como o jogo foi desenvolvido nem em que plataforma é executado. Como complemento, os dois jogos sem denominação de tipo pelos autores, pelas suas características de mover um personagem até seu objetivo usando-se de lógica, podem ser caracterizados também como *puzzle*, que são jogos com “estrutura lógica organizada e aberta que encaminha um processo reflexivo que culmina na compreensão de um dado problema que se constitui no próprio *puzzle*” (TONÉIS, 2016).

Para responder a segunda pergunta secundária, “Quais as características dos jogos sérios utilizados atualmente no ensino de lógica de programação?”, o jogo sério descrito no artigo de Silva et al. (2020) ensina conceitos iniciais de lógica de programação (de tipos de variáveis até estruturas de repetição) em um formato RPG onde, segundo os autores, “o jogador deve resolver quebra-cabeças, missões, entrar em batalhas e conversar com os personagens do jogo para que possam entender a história e o modo de jogo”. O jogo utiliza elementos 2D e visão em terceira pessoa. Pergaminhos encontrados nas fases e conversas com os personagens não jogáveis (NPCs) fornecem conhecimento sobre o assunto estudado, e durante o jogo o jogador deve resolver desafios que demonstrem a aquisição dos conhecimentos, como o desafio da Figura 1, onde o jogador deve escolher o caminho que contém um valor do tipo inteiro para prosseguir.

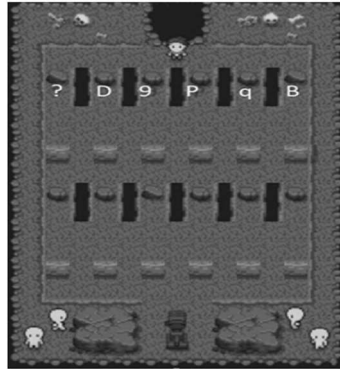


Figura 1. Desafio do jogo sério do artigo de Silva et al. (2020)

Ainda sobre o artigo de Silva et al. (2020), as batalhas comuns deste jogo seguem o modelo de turnos, um dos mais comuns em jogos RPG, mas para derrotar os chefes de cada fase é necessário que se responda questões sobre o assunto estudado na fase.

Já para o jogo do artigo de Tsai e Lai (2022), as características destacadas são a possibilidade de que, através da escolha de um tópico, o estudante possa ler conceitos do mesmo e entrar no ambiente RA para se aprofundar nos conhecimentos, onde após receber a descrição de um problema, o estudante cria códigos de programação em blocos em seu smartphone para solucioná-lo. Assim, há um processo autônomo do aprendizado. O problema, que aparece após a seleção de um tópico de conhecimento pelo jogador, é exibido em um cenário em RA pré-definido, onde o jogador deve mover algum objeto até um ponto final, concluindo assim o desafio e demonstrando o conhecimento. A Figura 2 mostra a imagem retirada de um smartphone com o mapa gerado após se escolher o primeiro tópico de conhecimento.

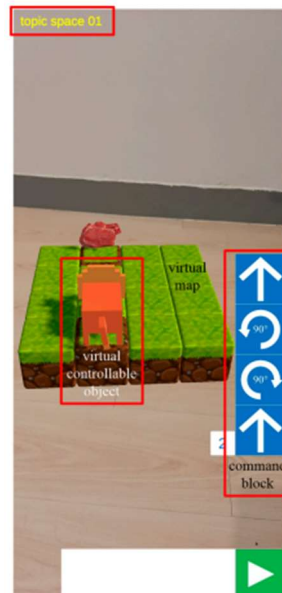


Figura 2. Imagem de uma fase do jogo de Tsai e Lai (2022)

Selecionando os blocos azuis à direita da tela, eles se deslocam para a área branca, e clicando no botão verde a ação do objeto pelo mapa é iniciada. Os conhecimentos abordados vão de estrutura lógica até estruturas de repetição.

O artigo de Lee e Quek (2018) é um jogo cujo objetivo é mover um personagem que deve recolher lixo “jogado” em certos locais do mapa e reciclá-lo, jogando na lixeira correspondente a seu tipo. A jogabilidade se caracteriza de forma similar ao anterior, bem como a distribuição dos mapas, também vista em um *smartphone* via RA, porém ao invés de se selecionar as ações do personagem a ser movido usando blocos pela tela do dispositivo, isso é feito através de *pendrives* conectados nas 4 portas USB de um Arduino. Há 5 tipos diferentes de *pendrive*, representando comandos de movimentação, captura e reciclagem do lixo, e a ordem em que são inseridos determina as ações do personagem. A Figura 3 mostra o hardware envolvido (*pendrives* e Arduino, em uma caixa), e um livreto que contém a história e o marcador RA para gerar o mapa no *smartphone*.



Figura 3. Hardware do jogo do artigo de Lee e Quek (2018)

Para iniciar a fase, o jogador coloca o livreto na caixa conforme a Figura 3 (acima), e com o HMD, visualiza o mapa gerado pelo marcador RA. Compreendendo a situação, o jogador insere na ordem desejada os *pendrives* correspondentes às ações que deseja que o personagem faça e aperta o botão Scan no Arduino, que lê os comandos pelas portas USB e envia ao HMD via Bluetooth.

Concluindo as características dos jogos, o artigo de Hong e Chu (2017) descreve que se situa em um ambiente virtual no espaço, cujo objetivo do jogador é usar blocos de comando de lógica de programação para manipular os robôs do jogo e escapar dos inimigos. Há 4 missões, cada uma com várias tarefas, que exigem conhecimentos de lógica de programação para serem realizadas. Os jogadores recebem a explicação e instrução das tarefas, e utilizam os blocos para realizar as ações necessárias. Se errarem, devem passar pela explicação e instrução novamente. Os conteúdos abordados vão de sequência lógica até estruturas de repetição. A Figura 4 mostra uma das telas do jogo.

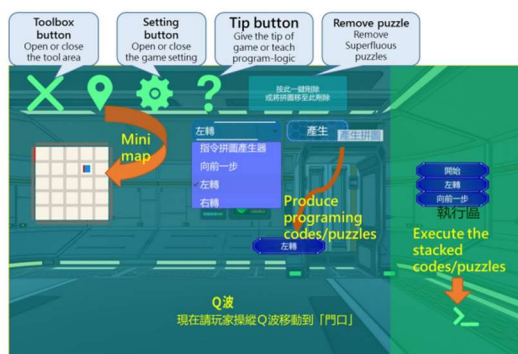


Figura 4. Imagem de uma fase do jogo de Hong e Chu (2017)

Por fim, se faz necessário apontar “Quais as métricas empregadas para avaliar o uso de jogos sérios no ensino de lógica de programação?”, respondendo assim a última questão secundária. No artigo de Silva et al. (2020), 19 estudantes se basearam na versão protótipo do jogo para responder a um questionário com perguntas pessoais sobre os jogadores e 9 perguntas relacionadas ao jogo, utilizando o modelo MEEGA+ de avaliação de qualidade de jogos educacionais. As respostas eram informadas em escala de 5 pontos, de discordo totalmente a concordo totalmente. O artigo de Silva et al. (2020) apresenta também a porcentagem de respostas em cada nível da escala para as 9 questões e converte a escala em valores de 1 a 5 para gerar o *Mean Opinion Score* (MOS) e classificar as respostas, considerando que um MOS de 4,4 a 5 representava excelência e 4,0 a 4,3 era um bom resultado.

Para avaliar o uso do jogo do artigo de Tsai e Lai (2022), foram usados 2 questionários: um de efetividade do aprendizado (aplicado antes e depois da atividade com o jogo) e; um de motivação de aprendizado, baseado no modelo ARCS (atenção, relevância, confiança e satisfação) proposto por John Keller. O primeiro era dividido em 4 seções: informações demográficas, integração da lógica de programação em situações reais, conceitos de lógica de programação e aplicação desses conceitos, sendo as 3 últimas testes. Já o segundo tinha 20 questões, 5 em cada aspecto do ARCS, a serem respondidas em escala Likert de 5 pontos. Os métodos usados na análise dos resultados foram, respectivamente, ANCOVA (análise de covariância) e ANOVA (análise de variância).

No artigo de Lee e Quek (2018), a avaliação foi feita por um questionário aplicado com 10 estudantes após o uso do jogo por 15 minutos, onde deveriam responder questões sobre a interface, a RA e experiência de aprendizado usando o jogo. Posteriormente, 10 representantes do Ministério da Educação local jogaram o jogo e responderam ao questionário. Os autores não explanam as questões, nem sua origem, apenas resumem as respostas em 3 pontos de vista: os que gostaram, os que não gostaram, e os que não conseguiram jogar por terem experienciado tontura usando o HMD.

O artigo de Hong e Chu (2017) usou como métrica de avaliação questionários diversos. Para avaliar a capacidade de solução de problemas com programação, um questionário baseado no trabalho realizado por Lai e Hwang em 2014 (porém, o mesmo não se encontra como referência do artigo e não pôde ser encontrado), onde cada dimensão é avaliada com 7 itens através de uma escala Likert de 5 pontos. Para avaliar a satisfação com a abordagem de aprendizado, foi usado como base o trabalho de Chu et al. (2010), que consiste em 7 itens também avaliados através de uma escala Likert de 5 pontos. Já a avaliação da carga cognitiva foi feita com o questionário desenvolvido por Sweller, van Merriënboer e Paas em 1998 (porém o mesmo também não se encontra como referência do artigo e não foi encontrado), que possui 8 questões, sendo 5 que verificam a carga mental e 3 que medem o esforço mental.

4. Discussão

Sobre os tipos de jogos sérios utilizados atualmente no ensino de lógica de programação, alguns aspectos diferentes podem ser discutidos: plataformas de criação, plataformas de execução e estilo do jogo.

Como plataformas de criação, Silva et al. (2020) e Lee e Quek (2018) utilizaram a plataforma Unity, uma das principais plataformas de programação de jogos. O artigo de Tsai e Lai (2022) teve seu jogo desenvolvido em ARCore, uma plataforma Google para

desenvolvimento de aplicativos em RA. O artigo de Hong e Chu (2017) não descreveu a plataforma de desenvolvimento utilizada. Plataformas complementares também foram utilizadas, como Tiled (usado para criação de fases e personagens do artigo de Silva et al. (2020)) e Vuforia (para os marcadores RA do artigo de Lee e Quek (2018)). Assim, é possível perceber a variação na escolha do modo de desenvolver jogos sérios para lógica de programação, e que esse não parece ter sido um aspecto importante para o sucesso do jogo. Nota-se, portanto, que os autores utilizaram recursos que levam a aprendizagem para o domínio do ambiente real acrescido da tridimensionalidade virtual (RA).

Sobre onde esses jogos são executados, o artigo de Silva et al. (2020) apresenta um jogo sério para computadores, os artigos de Tsai e Lai (2022) e Lee e Quek (2018) utilizam *smartphones* e o artigo de Hong e Chu (2017) não apresenta essas informações. O artigo de Lee e Quek (2018) ainda apresenta o uso de dispositivos físicos como Arduino Uno e *pendrives* que trabalham em conjunto com o *smartphone* para executar o jogo. Isso mostra uma tendência de uso de equipamentos mais comuns para execução dos jogos sérios no ensino de lógica de programação, mas também a possibilidade de intercambiar tecnologias físicas para alterar as dinâmicas de jogabilidade.

Quanto ao estilo dos jogos, dois artigos (Tsai e Lai (2022) e Lee e Quek (2018)) apresentam um estilo muito comum em cursos *online* infantis: o de movimentar algum objeto/personagem de um ponto a outro do mapa. Essa jogabilidade caracteriza esses jogos como do estilo *puzzle*, denominação utilizada também pelos autores do artigo de Hong e Chu (2017) para descrever seu jogo. Por fim, o artigo de Silva et al. (2020) traz um jogo de estilo RPG, um estilo que geralmente é visto em jogos não casuais, ou seja, jogos que exigem do jogador mais tempo de uso para progredir, mas também com mecânicas *puzzle* para a resolução de alguns desafios.

Quanto às características dos jogos sérios utilizados no ensino de lógica de programação, destaca-se a semelhança dos objetivos de aprendizado: todos trazem conceitos iniciais da lógica de programação: variáveis, estrutura sequencial, estruturas condicionais e de repetição. Em relação à jogabilidade, o artigo de Silva et al. (2020) é o que possui mais variações na dinâmica de jogo, tendo diferentes fases com objetivos e maneiras diferentes de se adquirir o conhecimento necessário para superar os obstáculos propostos e demonstrar a aquisição do conhecimento, diferentemente dos jogos descritos nos demais artigos, que possuem dinâmicas similares em todas as fases, apenas com dificuldade incremental a cada sucesso. Nos artigos de Silva et al. (2020), Lee e Quek (2018) e Hong e Chu (2017), também é visível a preocupação com a narrativa do jogo, e no artigo de Lee e Quek (2018) essa narrativa ainda traz ao jogador a possibilidade de outro aprendizado: o da coleta seletiva do lixo.

Ao buscar avaliar o uso de jogos sérios no ensino de lógica de programação, todos os artigos usaram questionários aplicados com os jogadores, de tamanhos e intenções variadas: no artigo de Silva et al. (2020) aparecem questões relacionadas ao jogo; no artigo de Tsai e Lai (2022) sobre efetividade e motivação do aprendizado; o artigo de Lee e Quek (2018) foca na experiência e tecnologia RA utilizada; e o artigo de Hong e Chu (2017) questiona 3 aspectos (capacidade de solução de problemas com programação, satisfação com a abordagem de aprendizado e carga cognitiva). Além disso, também houve variação nas métricas para avaliação dos resultados: no artigo de Silva et al. (2020), o MEEGA+, um modelo recente de avaliação de qualidade de jogos educacionais, foi utilizado em conjunto à técnica *Mean Opinion Score* (MOS) para avaliar a experiência; o

artigo de Tsai e Lai (2022) usou os modelos ANCOVA e ANOVA; já os artigos de Lee e Quek (2018) e Hong e Chu (2017) não apresentam claramente suas métricas, pois o artigo de Lee e Quek (2018) não explana a origem das questões e apenas discute suas respostas sem aplicar nenhuma fórmula ou modelo de análise e no artigo de Hong e Chu (2017), apesar ter usado 3 questionários baseados em trabalhos anteriores, só foi possível encontrar um desses trabalhos pois os demais não estavam como referências no mesmo. Assim, é possível notar que não há uma efetiva clareza quanto aos resultados das experiências, apesar das visões positivas apresentadas sobre os jogos desenvolvidos.

5. Conclusão

Esta pesquisa executou uma revisão sistemática da literatura sobre o uso de jogos sérios no ensino da lógica de programação, e através de três perguntas secundárias, aplicadas em artigos filtrados e selecionados que tratam o tema de 2013 até o presente momento, foi possível observar aspectos diferentes destes jogos.

Primeiramente, a baixa quantidade de artigos no tema aponta a possibilidade de evolução e discussão no uso de jogos sérios no ensino de lógica de programação.

Sobre os jogos encontrados, foi possível perceber que diferentes tipos de jogos foram desenvolvidos, para diferentes plataformas, em diferentes linguagens e com diferentes contextos. Então, de modo geral, não parece ter sido descoberta uma “fórmula secreta” para o desenvolvimento desses jogos, e sim que os jogos sejam criados de acordo com as ferramentas conhecidas/acessíveis pelos desenvolvedores dos projetos.

A semelhança entre os trabalhos se apresenta no conteúdo abordado: os conceitos iniciais de lógica de programação. Isso sugere que é nesse momento em que se encontra a maior necessidade de uso de técnicas diferenciadas de ensino, e que conforme os estudantes avançam, aprofundar-se em lógica é menos necessário do que o aprendizado de uma linguagem de programação. Ainda, a maioria das propostas utilizadas explora a tecnologia de Realidade Aumentada.

Em relação às métricas de avaliação dos jogos, nota-se que o foco de cada autor é diferente, então entender se o mais importante em relação ao jogo é a experiência do jogador, o aprendizado proporcionado, a jogabilidade do mesmo ou algum outro aspecto é algo que ainda pode ser melhor discutido.

Por fim, é importante destacar que a baixa quantidade de artigos pode ter ocorrido devido a não utilização de algum mecanismo de busca ou pela frase de busca selecionada. Assim, sugere-se como possível continuidade da pesquisa ampliar a quantidade de mecanismos de busca, bem como revisar a frase utilizada buscando diferentes termos que possam ter sido utilizados em outros artigos que também atenderiam ao que esse trabalho busca responder.

Agradecimentos

O terceiro autor gostaria de agradecer ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq-Brasil) pela bolsa de produtividade DT2, processo 313398/2019-4 e à Fundação de Amparo à Pesquisa e Inovação do Estado de Santa Catarina (FAPESC-Brasil) pelo financiamento parcial ao laboratório LARVA (*Laboratory for Research on Visual Applications*), T.O. No.: 2021TR851.

Referências

- Araújo, M. G., and Schwamborn, S. H. L. (2013). “A Educação Ambiental em análise SWOT”. *Ambiente & Educação*, 18(2), páginas 183-208.
- Borges, R. P. et al. (2018) "A Systematic Review of Literature on Methodologies, Practices, and Tools for Programming Teaching". *IEEE Latin America Transactions*, 16 (5), páginas 1468-1475.
- Buchinger, D., Cavalcanti, G. A. de S., and Hounsell, M. da S. (2014). “Mecanismos de busca acadêmica: uma análise quantitativa”. *Revista Brasileira de Computação Aplicada*, 6(1), páginas 108–120.
- Chu, H. C., Hwang, G. J., Tsai, C. C., and Tseng, Judy C. R. (2010). “A two-tier test approach to developing location-aware mobile learning systems for natural science courses”. *Computers & Education*, 55(4), páginas 1618-1627.
- Oliveira, T. de, Stringhini, D., and Corrêa, D. G. M. (2018). "Strategies Focused on the Teaching of Programming Logic: A Systematic Review of Brazilian Literature". In *XIII Latin American Conference on Learning Technologies (LACLO)*, páginas 292-298.
- Hong, T. and Chu, H. (2017). “Effects of a Situated 3D Computational Problem-Solving and Programming Game-Based Learning Model on Students' Learning Perception and Cognitive Loads”. In *6th IIAI International Congress on Advanced Applied Informatics*, páginas 596-600.
- Klock, A.C.T. (2018). “Mapeamentos e Revisões Sistemáticas da Literatura: um Guia Teórico e Prático”. *Cadernos de Informática*, 10 (1), páginas 01–09.
- Lee, K. T., and Quek, A. (2018). “TARogic: Tangible augmented reality game”. *International Journal of Engineering and Technology (UAE)*, 7(2.14), páginas 101-104.
- Maskeliūnas, R., Kulikajevs, A., Blažauskas, T., Damaševičius, R., and Swacha, J. (2020). “An Interactive Serious Mobile Game for Supporting the Learning of Programming in JavaScript in the Context of Eco-Friendly City Management”. *Computers*, 9(4), páginas 102-119.
- Silva, J. P., Silveira, I. L., Kamimura, L. and Barboza, A. T. (2020). “Turing Project: An Open Educational Game to Teach and Learn Programming Logic”. In *15th Iberian Conference on Information Systems and Technologies (CISTI)*, páginas 1-6.
- Tonéis, Cristiano N. (2016). “O design de Puzzles nos jogos digitais”. In *XV Simpósio Brasileiro de Jogos e Entretenimento Digital*, páginas 404-411.
- Tsai, C. and Lai, Y. (2022). “Design and Validation of an Augmented Reality Teaching System for Primary Logic Programming Education”. *Sensors*, 22(1), páginas 389-406.