
**A GAMIFICAÇÃO COMO ALTERNATIVA NA APRENDIZAGEM DE
PROGRAMAÇÃO**

**GAMIFICATION AS AN ALTERNATIVE IN THE LEARNING OF
SCHEDULE**

Amanda Mika Omuna¹
Tânia Camila Kochmansky Goulart²

RESUMO

A gamificação é uma abordagem que utiliza elementos de jogos em contextos não relacionados a jogos, como educação e treinamento. Atualmente, torna-se um desafio a busca por métodos e motivações para o engajamento do aluno diante do aprendizado, já que muitos alunos apresentam dificuldades em se concentrar. Com isso nos questionamos: como fazer com que os alunos se sintam estimulados a aprender e se mantenham interessados? Uma possível resposta poderia ser o uso do método de gamificação. Neste trabalho apresentamos uma pesquisa exploratória através do levantamento de dados sobre a aprendizagem de programação, realizando uma análise de softwares que possuem gamificação implementada e comparar suas sintaxes com linguagens de programação existentes. Nosso objetivo é investigar e discutir os efeitos do uso da gamificação e como esse método pode trazer benefícios à aprendizagem.

Palavras-chave: Ciência da Computação; gamificação; programação; educação digital.

ABSTRACT

Gamification is an approach that uses game elements in non-game contexts such as education and training. Currently, the search for methods and motivations for student engagement in learning becomes a challenge, since many students have difficulty concentrating. With that, we ask ourselves: how to make students feel encouraged to learn and stay interested? A possible answer could be the use of the gamification method. In this work we present an exploratory research through the collection of data on programming learning, performing an analysis of software that have implemented gamification and comparing their syntax with existing programming languages. Our goal is to investigate and discuss the effects of using gamification and how this method can bring benefits to learning.

Keywords: Computer Science; gamification; schedule; digital education.

¹ Discente do curso de Ciência da computação do Centro Universitário Filadélfia - UniFil.

² Docente do curso de Ciência da computação do Centro Universitário Filadélfia - UniFil.

1 INTRODUÇÃO

Com o passar do tempo é inegável que está ficando difícil de despertar a vontade de aprendizagem dos alunos, muitos não apresentam um bom desempenho escolar que, entre outras coisas, pode estar ligado ao ensino de aprendizagem tradicional, que por muitos é tachada de tediosa e/ou monótona. Na aprendizagem de programação, a gamificação pode ser uma estratégia eficaz para envolver e motivar os estudantes, tornando o processo de aprendizagem mais divertido e atraente.

Segundo Aleksic-Maslac, Rasic e Vranesic (2018), alunos são expostos à sobrecarga de informações todos os dias, então não é surpresa que cada vez fique mais desafiador ficar sentado durante todo o horário escolar e se concentrar nas palestras dos professores. Mas, por outro lado, *gamers* podem ficar sentados por horas sem perder uma grande quantidade de energia e interesse, já que jogos são divertidos e dinâmicos.

Diante dos fatos e problemas apontados, uma possível alternativa seria o uso do método de gamificação, tendo em vista o dinamismo atraído pelos jogos. Com isso, a gamificação apresenta-se como uma estratégia no ensino e aprendizagem.

O termo gamificação, por sua vez, vem do inglês *gamification* e se trata do uso de elementos de design de jogos em um contexto não lúdico, a fim de melhorar o engajamento e experiência do usuário (DETERDING *et al.*, 2011).

Nos últimos anos, possibilitou-se o rápido desenvolvimento de recursos e ferramentas digitais, como smartphones, computadores, aplicativos, dentre outros recursos que podem contribuir na inovação da aprendizagem dentro e fora da sala de aula.

Existem milhares de softwares educacionais disponíveis atualmente, mas a gamificação se destaca dentre os demais em questão de eficácia do ensino e na obtenção do engajamento do aluno no aprendizado. E com isso nos questionamos: como fazer com que os alunos se sintam estimulados a aprender e, ao mesmo tempo, se mantenham interessados por um tempo razoável? Uma possível resposta poderia ser o uso do método de gamificação, devido à existência do dinamismo presente também em jogos, que deixa o usuário empenhado e/ou vidrado.

Para tal hipótese, neste estudo faremos uma pesquisa realizada a partir da análise de obras acerca do assunto de gamificação, sendo analisado também os efeitos do uso da gamificação na aprendizagem de programação.

Será feita uma pesquisa descritiva, do tipo exploratória, havendo uma abordagem qualitativa acerca do método. Será realizado uma análise através da comparação de sintaxes entre linguagens de programação existentes e os softwares escolhidos, advindos da revisão bibliográfica, para um mesmo algoritmo elaborado. E por fim, a interpretação seguida pela conclusão final, onde traçaremos nossas considerações.

2 ALGUMAS LINGUAGENS

Na seção, apresentaremos uma abordagem teórica, construída mediante uma revisão bibliográfica a respeito do foco da nossa pesquisa.

Começamos com a pesquisa de Çubukçu *et al.* (2017), que aborda a inserção no ensino à linguagem Java. A pesquisa consiste no desenvolvimento do GeNIE, um sistema educacional com elementos da gamificação, a implementação da disciplina de Introdução ao Java no sistema e a aplicação do mesmo em um curso da University College Dublin durante um semestre, tendo como critério de avaliação a pontuação obtida.

Como a gamificação é uma área relativamente nova comparada a outras áreas de pesquisa, com o processo de desenvolvimento desse estudo foi possível evidenciar que essa área sofre de escassez de suporte de software necessário para ser aplicada adequadamente. Mas embora esse estudo tenha tido resultados sólidos bastante limitados, os estudantes demonstraram entusiasmo sobre o sistema e a ideia da gamificação.

Na pesquisa de Murad *et al.* (2018), foi realizado o desenvolvimento de um aplicativo para o ensino de língua estrangeira. Os participantes relataram uma experiência positiva com o aplicativo desenvolvido (o aplicativo SLIONS), apontando fatores como a facilidade de uso, diversão e potencial educacional, além da percepção da melhoria da própria pronúncia.

A pesquisa envolve uma ferramenta com gamificação chamada Kahoot, usada para aprendizagem repetitiva e preparações para exames.

Na pesquisa de Dalla Vecchia e Maltempi (2019), utilizou-se da metodologia qualitativa para aplicar um curso de construção de jogos eletrônicos em um curso de Licenciatura em Matemática a fim de apresentar o problema no processo de Modelagem Matemática envolvido na construção.

No curso ministrado, intitulado "Construção de Jogos Eletrônicos", utilizou-se do software *Scratch*, que apresenta uma linguagem de programação visual e os comandos são

representados por meio de blocos, os quais são movidos e conectados uns com os outros, formando a programação do ambiente. Participaram oito alunos de graduação de um curso de Licenciatura em Matemática, os quais participaram de 4 encontros no período de maio a julho de 2009 para a realização das atividades, bem como gravações.

Com os dados coletados do desenvolvimento dos jogos, foi possível analisar que, ao longo do processo de Modelagem Matemática, via dúvidas, perguntas e afirmações que conduzem às soluções para o problema, a imersão no problema vai desmembrando-o em determinações mais específicas. Isso pode ser mostrado com um fluxo elaborado na pesquisa para apresentar o processo da busca por uma solução.

Coore e Fokum (2019) realizaram a pesquisa consiste na introdução do uso de uma plataforma competitiva de programação como um mecanismo para autoavaliação para um curso introdutório sobre design e análise de algoritmos. A introdução tinha como objetivo aumentar o número de exercícios de programação avaliados para uma média de 1 por semana, visto que poucas tarefas eram aplicadas e o tempo que os alunos esperavam pelo feedback era longo.

Na pesquisa de Barata *et al.* (2013), foi realizado a aplicação da gamificação em um curso universitário via sistemas de desafios, tabelas de classificação, pontos de experiência, níveis e emblemas. A pesquisa visa explorar como a gamificação pode ser aplicada na educação para a melhoria do envolvimento do aluno.

O artigo de O'Donovan, Gain e Marais (2013) apresenta a aplicação da gamificação em um curso de Desenvolvimento de Jogos de Computador na University of Cape Town (UCT), utilizando um ambiente de aprendizagem online, também abordando como isso pode ser generalizado para outros cursos. O objetivo do estudo é melhorar a participação em palestras, a compreensão do conteúdo, o engajamento e a habilidade em resolução de problemas. A medida de avaliação usada foi através das notas de curso, avaliações dos palestrantes, presença em aula e um questionário elaborado.

Durante nosso processo de busca e leitura em artigos sobre o uso da gamificação para o ensino, nossos olhares estavam voltados para a seguinte indagação: Como a gamificação pode favorecer a aprendizagem de programação? Já que as pesquisas realizadas nos artigos apresentaram resultados satisfatórios para suas áreas de estudo, seria relevante falar da gamificação relacionada à programação e a importância dessa relação.

Sendo assim, estabelecida a nossa questão de pesquisa, temos como objetivo investigar como a gamificação pode contribuir para a programação, explanando e delineando os caminhos

seguidos por nós na construção da nossa problemática.

2.1 DESIGN E LINGUAGEM

E tendo em vista o uso satisfatório de sistemas como o Scratch na aprendizagem, segundo artigos da seção de revisão bibliográfica deste trabalho, realizamos uma busca e seleção por sistemas desenvolvidos para o ensino da programação. Para a análise neste trabalho consideraremos os seguintes softwares: Scratch, Code.org, RoboMind e HackerRank.

3 DESENVOLVIMENTO

A seguir, apresentaremos o design e a linguagem dos softwares escolhidos bem como os aspectos cognitivos que os envolvem.

3.1 SCRATCH

O Scratch é uma linguagem de programação baseada em blocos, desenvolvida no Massachusetts Institute of Technology (MIT). Esse software é usado para a introdução no estudo de programação dos alunos dos cursos de Ciência da Computação em geral.

É uma ferramenta orientada para a aprendizagem da programação de forma visual, interativa e lúdica, tornando compreensível o porquê de ser popularmente usado no ensino infantil.

Tal conceito é evidenciado na descrição de Marques, Levitt e Nixon (2012) que destacam que o uso do Scratch "oferecem um meio altamente gráfico e interativo de ensinar habilidades básicas de programação de computadores. Esse estilo de apresentação e interação é semelhante ao dos jogos digitais."(MARQUES; LEVITT; NIXON, 2012, p.15).

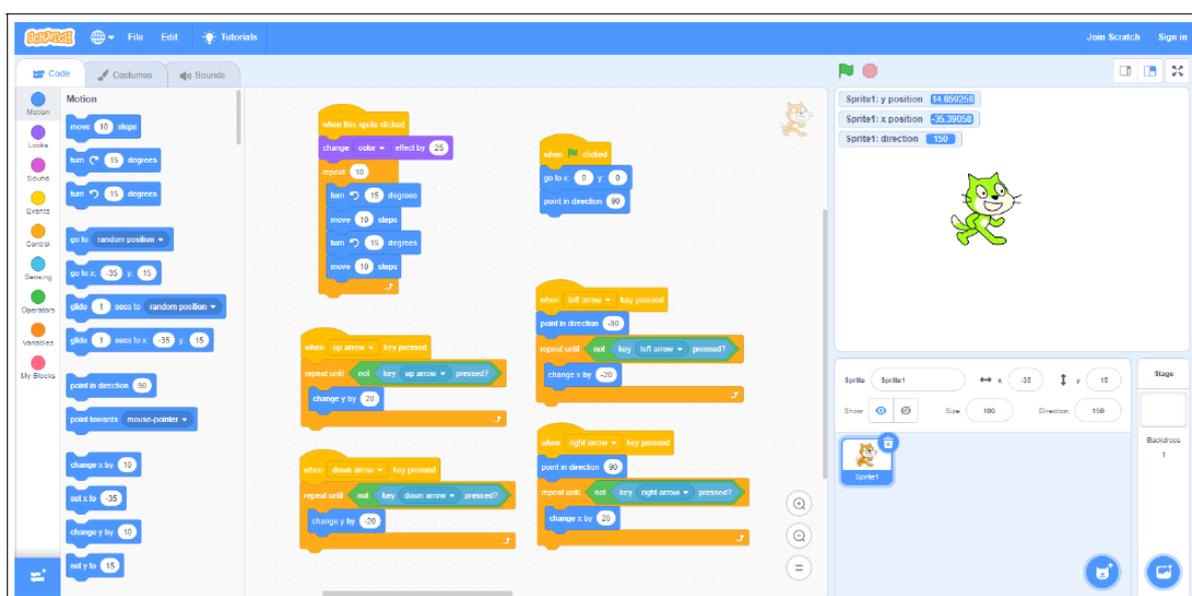
Partindo dessas afirmações, considera-se que a diferença entre o uso do Scratch e o de uma linguagem de programação mais avançada é de que o ensino se dá através da estrutura de blocos presente no Scratch. Nesse sentido, a ferramenta foca-se mais na lógica de programação pela simplicidade e facilidade de aprendizagem, enquanto uma linguagem de programação como o Python requer mais conhecimento em relação à sintaxe e suas correções.

Em relação ao design da página principal do site, ela consiste-se em uma bandeja de

blocos no canto esquerdo, divididos por funcionalidades, conforme está indicado na Figura 1.

Logo acima estão localizadas seções para os comandos em bloco, edição de sprites ou “atores” e edições sonoras. No centro situa-se o espaço para a elaboração do código através da montagem de blocos, arrastando os mesmos até o espaço. E no lado direito situa-se um display para a demonstração visual da execução do código e gerenciamento dos sprites e cenários que, neste caso, é mostrado apenas o personagem mascote da plataforma, e logo abaixo do display é onde se faz a seleção e alterações de atributos dos atores e cenários.

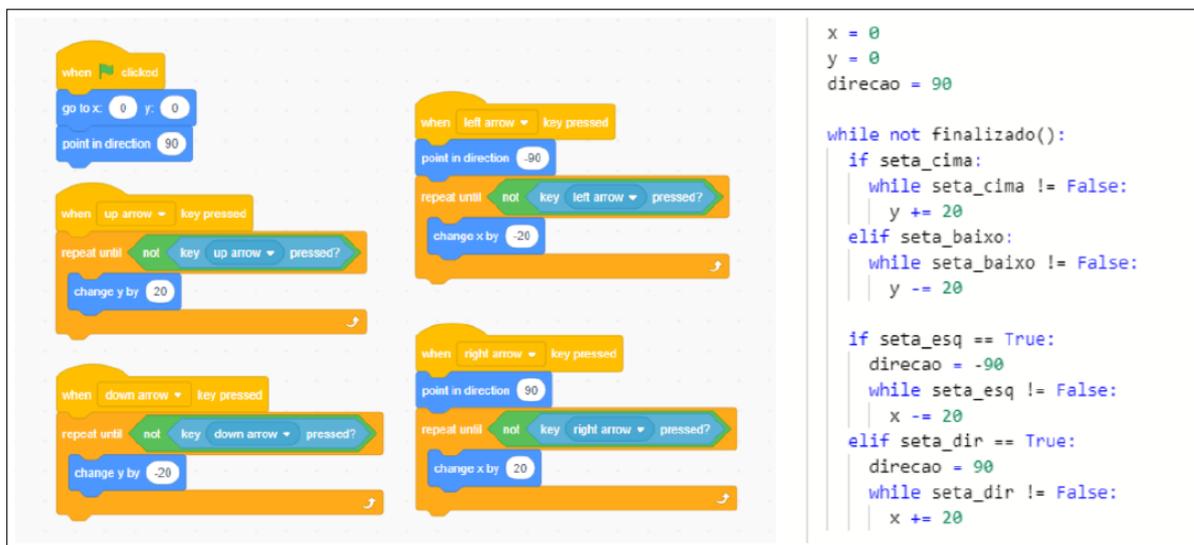
Figura 1 – Ambiente visual do Scratch



Fonte: Azevedo (2011)

Neste trabalho, apresentaremos, a título de comparação, uma figura (Figura 2) que apresenta o mesmo algoritmo indicado no Scratch e no Python, conforme a diferença de sintaxes citada anteriormente.

Figura 2 – Comparação entre Scratch e Python



Fonte: As autoras.

No lado direito da Figura 2, é apresentado um algoritmo desenvolvido no Python e no lado esquerdo, o mesmo algoritmo desenvolvido no Scratch. No lado esquerdo, é identificável o sistema de comandos por blocos, podendo realizar a montagem dos mesmos para elaborar seu próprio código. Percebe-se também que os blocos são organizados através de suas cores e, por consequência, suas funções.

Como os blocos azuis que, conforme a própria aplicação, são blocos de movimento que, comparando com a sintaxe do Python, estão relacionados a atribuições. E os blocos laranjas que são blocos de controle que geralmente envolvem comandos de loops, retratados na comparação como os laços while no Python.

Portanto, é evidenciado que o Scratch foca-se mais na lógica devido sua sintaxe ser mais fácil e simples comparada ao do Python.

3.2 CODE.ORG

O Code.org é uma organização sem fins lucrativos fundada por Hadi Partovi e Ali Partovi (empresários e investidores da cidade de Teerã, no Irã), que possui sua plataforma visando a expansão do ensino de programação a escolas e grupos subrepresentados.

Sua linguagem de programação visual assemelha-se a do Scratch, então também não é necessário o conhecimento de sintaxe de programação devido ao seu esquema simples de

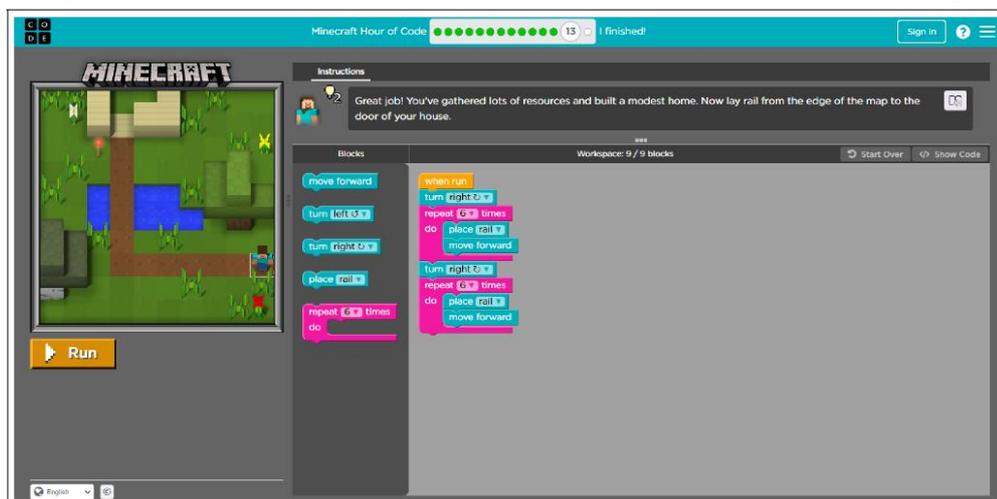
comandos em blocos, além de materiais e bibliotecas embutidos.

O sistema dispõe-se de diversos cursos com temas variados, podendo haver uma entrega de um certificado do site dependendo do curso.

A página principal de um desafio de um curso, neste caso o curso Minecraft Adventurer, é apresentado na Figura 3. Ela constitui-se de um display, uma tela visual, no canto esquerdo da tela para visualização do problema e execução do código, uma bandeja dos blocos disponíveis mais ao centro, um espaço para a montagem dos blocos. E mais acima, uma instrução do problema, podendo haver solicitação de dicas e a barra de progresso do curso.

A Figura 3 exibe a página principal de um desafio apresentada no curso Minecraft Adventurer.

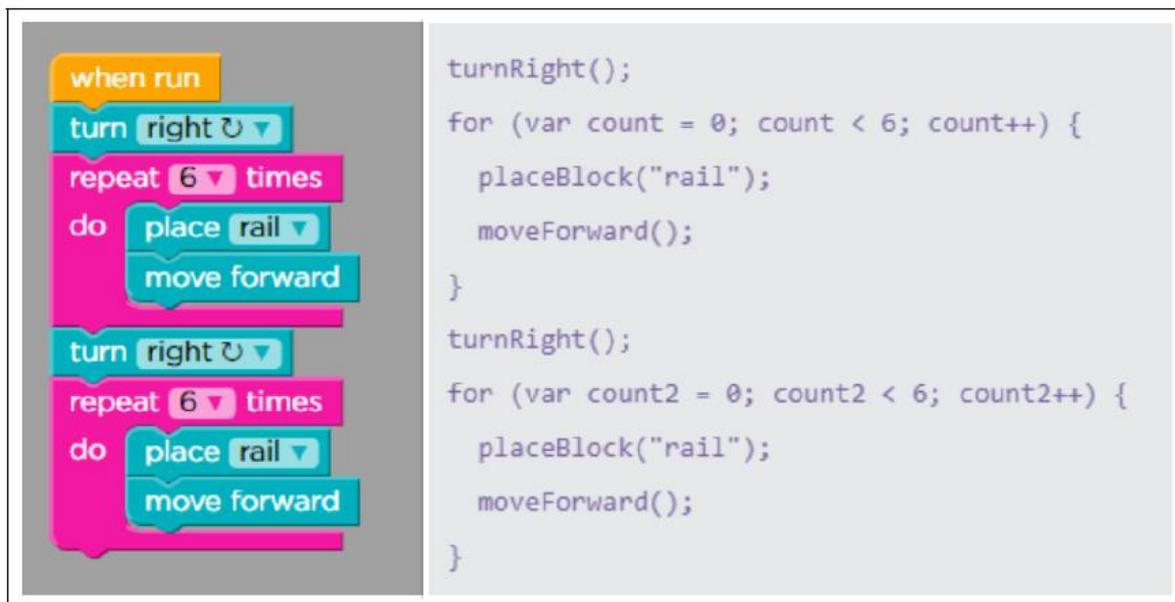
Figura 3 – Ambiente visual do curso Minecraft Adventurer no Code.org



Fonte: As autoras.

No final de cada desafio após a conclusão destes é possível visualizar uma versão gerada do código da resposta em JavaScript (Figura 4), disponível para análise e comparações, sendo sua visualização opcional.

Figura 4 – Ambiente visual do curso Minecraft Adventurer no Code.org



Fonte: As autoras.

O Code.org é uma plataforma de aprendizagem de programação voltada para iniciantes, enquanto JavaScript é uma linguagem de programação utilizada para desenvolvimento web. Code.org é uma introdução acessível e simplificada à programação, enquanto JavaScript é uma linguagem mais avançada e versátil com uma variedade de aplicações na web. Essa comparação de sintaxes também se assemelha à Figura 2, onde o software destaca-se no ensino à lógica de programação.

9

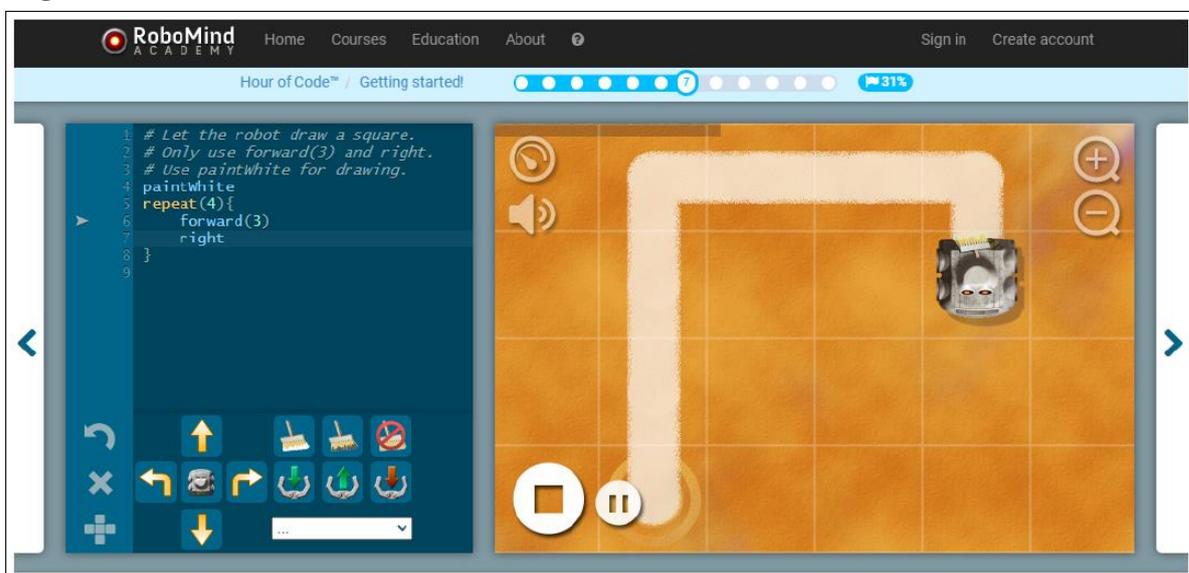
3.3 ROBOMIND

O RoboMind é um ambiente de programação desenvolvido por Arvid Halma, na Universidade de Amsterdam, que visa ensinar os processos básicos de programação através da simulação de um robô.

Este sistema se difere dos outros dois citados anteriormente devido à sua linguagem de programação textual que mais se assemelha às linguagens usadas no desenvolvimento de aplicações reais. Portanto, é esperado para esse tipo de linguagem que o estudante futuramente esteja mais apto e tenha facilidade ao avançar para uma linguagem de programação mais profissional.

A Figura 5 apresenta uma página de um desafio do curso Hour of Code. No lado esquerdo localiza-se um editor onde pode-se elaborar os códigos através dos botões no painel ou digitando. No lado direito há uma tela visual para representar a situação inicial e a execução do código e na parte de cima da página há a barra de progresso do curso. Esses desafios podem envolver a programação de robôs virtuais para realizar tarefas específicas, como mover-se, virar, pegar objetos, evitar obstáculos, entre outros. Os alunos podem usar a linguagem de programação visual do RoboMind, que consiste em arrastar e soltar blocos de comandos para criar sequências de ações para os robôs.

Figura 5 – Ambiente visual do curso Hour of Code no RoboMind



Fonte: As autoras.

Um dos principais diferenciais do software é que ele conta com seu próprio script de linguagem, introduzindo o usuário a técnicas de programação de forma simplificada. A Hour of Code é uma iniciativa global que busca introduzir conceitos básicos de programação a alunos de todas as idades. RoboMind é uma plataforma de aprendizagem de programação que utiliza uma linguagem de programação visual para programar robôs virtuais.

A Hour of Code no RoboMind é uma atividade que pode ser realizada para fornecer aos alunos uma experiência prática e interativa de programação. Durante essa atividade, os alunos são apresentados ao ambiente de programação do RoboMind e são guiados por uma série de desafios de programação.

A Figura 6 ilustra a comparação do script próprio do RoboMind com o da linguagem

Python.

Figura 6 - Comparação entre RoboMind e Python



Fonte: As autoras.

Com base na comparação entre a Hour of Code no RoboMind e o uso do Python, podemos destacar algumas diferenças e semelhanças.

O RoboMind utiliza uma linguagem de programação visual, onde os alunos arrastam e soltam blocos de comandos para criar programas para os robôs virtuais. Além disso, a linguagem visual do RoboMind simplifica a programação e é adequada para iniciantes. Os alunos podem se familiarizar com os conceitos básicos de programação por meio de uma interface gráfica intuitiva.

O Python é uma linguagem de programação textual amplamente utilizada. Os alunos escrevem código em Python usando uma sintaxe específica. E ainda, o Python é uma linguagem textual e pode ser considerada mais complexa para iniciantes. Requer compreensão da sintaxe e da estrutura da linguagem.

4 DISCUSSÃO DOS RESULTADOS

Os motivos pelos quais utilizamos a pesquisa exploratória como opção metodológica neste trabalho foi devido sua finalidade de levantar informações acerca de um assunto existente, não havendo a necessidade de se tratar ou realizar novas descobertas.

Neste caso se deu através do levantamento bibliográfico sobre a gamificação e

aplicações na aprendizagem de programação.

Foi feito então o levantamento bibliográfico do tema, pesquisado nos bancos digitais de artigos acadêmicos, e estabelecido a questão de pesquisa de “Como a gamificação pode favorecer a aprendizagem de programação?”. Para responder à questão, utilizamos os dados levantados da análise de design e linguagem dos softwares que possuem gamificação e da comparação dos mesmos com linguagens de programação mais avançadas.

Nos artigos acadêmicos listados no levantamento bibliográfico utilizaram-se de softwares para a realização de suas pesquisas empíricas, visando o ensino e aprendizagem, tais como o Scratch, HackerRank, Kahoot, entre outros. Dentre os resultados para essas pesquisas é possível apontar diversão, imersão, motivação e satisfação por parte dos usuários. É ressaltado também que essa área é relativamente nova em comparação com outras áreas de pesquisa.

Visando esses resultados, analisamos alguns desses softwares utilizados em questão de design e linguagem, apontando também como a gamificação se faz presente.

O design e linguagem entre os softwares partilham da mesma característica de apresentarem uma sintaxe mais simples e fácil, dado ao fato de possuírem programações visuais e lúdicas, e o foco na lógica de programação, com exceção do HackerRank. Além disso, em sua maioria utilizam-se de um sistema de recompensas, sendo por meio de pontos, emblemas ou diplomas, obtidos através da conclusão de desafios ou tarefas.

As últimas características se relacionam com Schlemmer (2014) quando o mesmo explica que a gamificação pode ser pensada a partir da perspectiva de persuasão, estimulando a competição via sistemas de pontuação, recompensas, etc. Além da perspectiva de construção colaborativa e cooperativa, incentivada pelas missões, desafios, descobertas e empoderamento em grupo.

Complementando o conceito de gamificação, Deterding *et al.* (2011) afirmam que as mecânicas de jogos que a gamificação possui tem o foco de melhorar o engajamento dos usuários através dos dados estatísticos das pesquisas. E também com a pesquisa de Coore e Fokum (2019), onde relatam um alto engajamento dos participantes. Não só esses efeitos como também a participação em fóruns e questionários também foi levantado, conforme as pesquisas de O’Donovan, Gain e Marais (2013), Barata *et al.* (2013) e Hew *et al.* (2016).

Além disso, nas pesquisas de O’Donovan, Gain e Marais (2013) e Çubukçu *et al.* (2017) é ressaltado pelos usuários que as tabelas de classificações é o fator mais motivador, enquanto os emblemas são o fator menos motivador.

Nesse sentido e em concomitância com os autores aqui citados, há uma melhor absorção do conhecimento devido a experiências memoráveis retidas pela interatividade e ludicidade.

Uma desvantagem relacionada à gamificação na aprendizagem de programação no nosso entendimento é a necessidade de arcar com os custos monetários e de tempo para uma melhor aplicação e implementação da gamificação (O'DONOVAN; GAIN; MARAIS, 2013). E também essa área é considerada relativamente nova em comparação com outras (ÇUBUKÇU *et al.*, 2017). Além disso, há o fato do método nem sempre traz os mesmos efeitos para todos os envolvidos devido as suas diferenciações individuais, segundo Coore e Fokum (2019) e Auvinen, Hakulinen e Malmi (2015) e a experiência do usuário. Isso relaciona-se com as assertivas de alguns dos resultados do levantamento bibliográfico, como as pesquisas de Aleksic-Maslac, Rasic e Vranesic (2018) e Barata *et al.* (2013), nas quais apresentam motivação e satisfação

5 CONSIDERAÇÕES FINAIS

13

O processo de desenvolvimento deste trabalho proporcionou a compreensão de como os efeitos da gamificação podem apoiar o ensino de programação. Com isso, pôde-se levantar a possibilidade de tratar desse assunto se aprofundando mais na área computacional.

Com isso, listamos alguns dos principais efeitos que o uso de ferramentas com gamificação incorporada proporciona: o envolvimento e maior participação nas atividades, o engajamento, a satisfação e, principalmente, a motivação em relação à aprendizagem. São apontados como os elementos mais motivadores as tabelas de classificação e os menos motivadores, os emblemas. Quanto ao design e linguagem analisados, é bastante presente as questões visuais, o uso de linguagens visuais e a visualização do passo-a-passo dos comandos, havendo uma facilidade no uso.

Com a importância da aprendizagem de programação para a área educacional, dito ser uma das competências que visam sustentar os pilares da educação, os efeitos que a gamificação traz consigo são benéficos à aprendizagem de programação.

Suas mecânicas de jogos são o que estimulam as cooperatividade e/ou competitividade entre os alunos, e os designs e linguagens visuais trazem consigo uma experiência memorável por serem lúdicas e interativas, possibilitando uma melhor absorção e retenção do

conhecimento. Sendo assim, a questão de pesquisa é respondida ao concluir que a junção do uso das mecânicas de jogos, presentes na gamificação, e os aspectos de design das ferramentas trazem um melhor resultado à aprendizagem, comparado métodos tradicionais.

Dessa forma, espera-se que este trabalho contribua para as possíveis futuras pesquisas sobre o tema, dada a importância do assunto e tendo em vista o fato da gamificação ser um termo relativamente novo. Além da dificuldade de tornar os efeitos da gamificação mais uniformes para todos os envolvidos. Há a necessidade da ampliação de seu desenvolvimento e formas de aplicação da gamificação para o aproveitamento da área educacional, computacional e de outras áreas. É importante ressaltar que a gamificação deve ser usada como uma estratégia complementar dentro de um currículo bem estruturado de programação. Ela não substitui a compreensão dos conceitos fundamentais e a prática real de programação, mas pode aumentar o engajamento e a motivação dos alunos durante o processo de aprendizagem. Além disso, devemos ressaltar que a escolha de uma abordagem e aplicação, depende do objetivo, do nível de experiência dos alunos e dos recursos disponíveis.

REFERÊNCIAS

14

ALEKSIC-MASLAC, K.; RASIC, M.; VRANESIC, P. Influence of gamification on student motivation in the educational process in courses of different fields. *In: INTERNATIONAL CONVENTION ON INFORMATION AND COMMUNICATION TECHNOLOGY, ELECTRONIC AND MICROELECTRONICS (MIPRO)*, 41., 2018, Opatija, Croatia. **Proceedings** [...] Opatija, Croatia: IEEE, 2018.

AUVINEN, T.; HAKULINEN, L.; MALMI, L. Increasing students' awareness of their behavior in online learning environments with visualizations and achievement badges. **IEEE Transactions on Learning Technologies**, [S.l.], v. 8, n. 3, p. 261–273, 2015.

BARATA, G. *et al.* **Improving participation and learning with gamification**. *In: CONFERENCE ON GAMEFUL DESIGN, RESEARCH, AND APPLICATIONS*, 1., 2023, Nova Iorque, EUA. **Proceedings** [...]. Nova Iorque, EUA: Association for Computing Machinery, 2013. p. 10–17. Disponível em: <https://doi.org/10.1145/2583008.2583010>. Acesso em: 25 jul. 2023.

COSTA, S. S. *et al.* Um estudo exploratório dos games para introdução ao pensamento computacional. *In: CONGRESSO NACIONAL DE AMBIENTES HIPERMÍDIA PARA APRENDIZAGEM*, 7., 2015, São Luís. **Anais** [...] . São Luís: CONAHPA, 2015.

ÇUBUKÇU, Ç. *et al.* Gamification for teaching java. *In: CONFERENCE ON SIMULATION TOOLS AND TECHNIQUES*, 10., 2017, [S.l.]. **Proceedings** [...]. [S.l.]: ACM, 2017.

DALLA VECCHIA, R.; MALTEMPI, M. V. O problema na modelagem matemática: determinação e transformação. **Bolema**: Boletim de Educação Matemática, [S.l.], v. 33, n. 64, p. 748–767, ago. 2019.

O'DONOVAN, S.; GAIN, J.; MARAIS, P. **A case study in the gamification of a university-level games development course**. New York, NY, USA: Association for Computing Machinery, 2013.