

Bora Jogar: desenvolvimento de jogos para auxiliar na aprendizagem de programação

Let's Play: game development to aid in programming learning

Juguemos: desarrollo de juegos para ayudar en el aprendizaje de la programación

Recebido: 04/03/2022 | Revisado: 12/03/2022 | Aceito: 26/03/2022 | Publicado: 01/04/2022

Ana Paula Berto Bezerra

ORCID: <https://orcid.org/0000-0002-9320-615X>

Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, Brasil

E-mail: ana.berto@academico.ifpb.edu.br

Ewerthon Dyego de Araújo Batista

ORCID: <https://orcid.org/0000-0003-4993-9900>

Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, Brasil

E-mail: ewerthon.batista@ifpb.edu.br

Jamile Farias de Sousa

ORCID: <https://orcid.org/0000-0001-5998-660X>

Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, Brasil

E-mail: jamile.farias@academico.ifpb.edu.br

Taynná de Almeida Olegário

ORCID: <https://orcid.org/0000-0001-8278-2103>

Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, Brasil

E-mail: taynna.olegario@academico.ifpb.edu.br

Eliel Bento Evangelista

ORCID: <https://orcid.org/0000-0002-6451-150X>

Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, Brasil

E-mail: eliel.bento@academico.ifpb.edu.br

Romeryto Vieira Lira

ORCID: <https://orcid.org/0000-0003-2567-0839>

Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, Brasil

E-mail: romeryto.lira@academico.ifpb.edu.br

Resumo

Introdução: a desistência de alunos sempre foi um problema na área da tecnologia da informação, e uma das principais causas que influenciam é a dificuldade que os estudantes apresentam em aprender programação. Visando amenizar esse problema, o presente artigo se baseia na gamificação para aprimorar e facilitar o estudo de programação através da criação de jogos de tabuleiro e digitais. **Objetivo:** criar jogos de tabuleiro e digitais para auxiliarem no ensino e no aprendizado de programação para alunos iniciantes. **Metodologia:** o projeto foi dividido em quatro fases: busca de artigos relacionados à utilização de jogos e a facilitação do aprendizado de alunos na programação; logo após, a partir das respostas de um *survey*, desenvolvido anteriormente, os temas usados para criar os jogos foram escolhidos, houve a implementação da versão em tabuleiro e da versão digital dos jogos. **Resultados:** a partir da busca de materiais que tivessem relação com o tema, foram selecionados seis artigos relevantes para análise, de acordo com os critérios de inclusão e exclusão pré-determinados; a partir dos resultados da pesquisa que foi produzida com o intuito de detectar as maiores dificuldades dos alunos na disciplina de Algoritmos e Lógica de programação, foram obtidas as respostas que ajudaram a nortear a produção dos jogos. As versões em tabuleiro foram produzidas com o draw.io, um editor de gráficos online, e as versões digitais foram desenvolvidas com a ferramenta Flippity, usada para criar diferentes tipos de jogos.

Palavras-chave: Jogos; Aprendizagem; Ensino; Programação.

Abstract

Introduction: student dropout has always been a problem in the area of information technology, and one of the main causes that influence it is the difficulty that students have in learning programming. In order to alleviate this problem, this article is based on gamification to improve and facilitate the study of programming through the creation of board and digital games. **Objective:** to create board and digital games to assist in teaching and learning programming for beginning students. **Methodology:** the project was divided into four phases: search for articles related to the use of games and the facilitation of student learning in programming; soon after, from the answers of a survey, developed previously, the themes used to create the games were chosen, there was the implementation of the board version and the digital version of the games. **Results:** from the search for materials that were related to the theme, six relevant articles were selected for analysis, according to the pre-determined inclusion and exclusion criteria; from the results of the

research that was produced with the aim of detecting the biggest difficulties of the students in the discipline of Algorithms and Logic of programming, the answers were obtained that helped to guide the production of the games. The board versions were produced with draw.io, an online graphics editor, and the digital versions were developed with the Flippity tool, used to create different types of games.

Keywords: Games; Learning; Teaching; Programming.

Resumen

Introducción: la deserción estudiantil siempre ha sido un problema en el área de tecnologías de la información, y una de las principales causas que inciden en ella es la dificultad que tienen los estudiantes para aprender a programar. Para paliar este problema, este artículo se basa en la gamificación para mejorar y facilitar el estudio de la programación a través de la creación de juegos de mesa y digitales. **Objetivo:** crear juegos de mesa y digitales para ayudar en la enseñanza y el aprendizaje de la programación para estudiantes principiantes. **Metodología:** el proyecto se dividió en cuatro fases: búsqueda de artículos relacionados con el uso de juegos y la facilitación del aprendizaje de los estudiantes en programación; luego, a partir de las respuestas de una encuesta, desarrollada previamente, se eligieron los temas utilizados para la creación de los juegos, se procedió a la implementación de la versión de tablero y la versión digital de los juegos. **Resultados:** a partir de la búsqueda de materiales que tuvieran relación con el tema, fueron seleccionados seis artículos relevantes para el análisis, de acuerdo con los criterios de inclusión y exclusión predeterminados; a partir de los resultados de la investigación que se produjo con el objetivo de detectar las mayores dificultades de los estudiantes en la disciplina de Algoritmos y Lógica de programación, se obtuvieron las respuestas que ayudaron a orientar la producción de los juegos. Las versiones de tablero se produjeron con draw.io, un editor de gráficos en línea, y las versiones digitales se desarrollaron con la herramienta Flippity, utilizada para crear diferentes tipos de juegos.

Palabras clave: Juegos; Aprendizaje; Enseñando; Programación.

1. Introdução

Atualmente, os cursos da área de Tecnologia da Informação (Ciência da computação, bacharelado em Engenharia de Software, Desenvolvimento de Sistemas e etc) apresentam um dos maiores índices de evasão e reprovação de alunos (de Carvalho et al., 2018). Por meio de técnicas de Mineração de Dados e de Inteligência Artificial, pesquisadores apontam que o baixo rendimento nas disciplinas iniciais do curso, principalmente, Introdução à Programação e Algoritmos, um dos principais ofensores para a evasão (Gonçalves et al., 2018) (Guerra et al., 2018).

Alunos iniciantes, ou com pouca experiência em Computação, são os que mais sofrem com programação. Os conceitos de matemática, de lógica e de habilidades para interpretar e para resolver problemas são pouco, ou quase nunca, trabalhados no ensino básico. Com isso, os alunos chegam aos cursos de Tecnologia da Informação (TIC) com uma base ruim ou inexistente. Complementar aos pontos levantados, programação é uma disciplina extremamente prática. Segundo Paiva et al. (2020), alunos inexperientes se assustam com a quantidade de habilidades exigidas para a resolução dos problemas computacionais que só serão adquiridos com muita prática. Consequentemente, ao se depararem com tantos desafios e falhas iniciais, os alunos perdem a motivação no estudo da disciplina. Uma possível solução para isso, segundo Paiva, seria o aumento da carga horária das aulas práticas para as disciplinas de programação.

De acordo com Zapata et al. (2020), a taxa de abandono dos cursos de TIC varia entre 40% e 60% dos matriculados. Esse número alto repercute diretamente nos setores da economia, da educação, ocasionando, diretamente, impacto social. Na contramão da evasão, o mercado do software está cada vez mais aquecido e necessitado de profissionais. Zapata afirma ainda que na Europa e nos Estados Unidos há, respectivamente, escassez de 800.000 a 1.000.000 profissionais na área de Computação.

Conforme o trabalho de Amorim et al. (2016), os alunos apresentaram certa dificuldade no aprendizado no curso de Algoritmos e Lógica de Programação, resultando na desistência e na desmotivação dos estudantes. Uma das formas encontradas para o estudo ser mais descontraído e divertido, e por conseguinte, os discentes terem mais motivação e um melhor aprendizado, seria o desenvolvimento de jogos digitais para diminuir os índices de reprovação e de abandono do curso. Por meio da

gamificação, esses objetivos foram concretizados, pois os alunos apresentaram mais interesse em participar das aulas e houve um crescimento nos índices de aprovação.

Como apontado por Muñoz et al. (2019), a gamificação vem sendo aplicada tanto na educação quanto na indústria, podendo ser definida como o uso de elementos de jogos (desafios, metas, reconhecimentos, feedbacks) na resolução de problemas, a melhora do engajamento das pessoas e o auxílio do ensino. Na educação, objeto de estudo deste projeto, o emprego da gamificação vem melhorando os índices escolares, motivando os alunos a estudarem, tornando as disciplinas mais atrativas e mais agradáveis para os alunos (Puerta et al., 2020) (Jácome-Amores et al., 2021).

Neste cenário, o objetivo deste projeto foi, utilizando gamificação, criar três jogos de tabuleiro e três jogos digitais para facilitar o estudante a criar, a testar e a refinar algoritmos. Ademais, os jogos visam auxiliar no ensino e no aprendizado da disciplina de Algoritmos e Lógica de Programação. Com a utilização dos jogos desenvolvidos, pretende-se diminuir as barreiras e os déficits de formação, tornar a disciplina mais atrativa e, por conseguinte, melhorar o desempenho dos alunos.

2. Fundamentação Teórica

A lógica de programação é a forma como são escritas instruções em uma sucessão de passos, ou seja um algoritmo, de uma forma com que o computador possa interpretar. Segundo Salvetti (Salvetti, 1999), um algoritmo é uma sequência finita de instruções ou operações, cuja execução, em tempo finito, resolve um problema computacional, qualquer que seja sua instância. A lógica de programação auxilia na escrita de um código, dessa forma, faz com que o computador entenda e interprete as instruções passadas por meio de uma linguagem de programação. Aprender lógica de programação é o passo inicial para aprender a programar.

O aprendizado da disciplina de algoritmo e lógica de programação pode não ser tão simples para a maioria dos estudantes, por isso técnicas estão sendo utilizadas para facilitar o aprendizado, uma delas é a implementação de gamificações. Por meio da gamificação é possível tornar conteúdos densos em materiais mais simples, interativos e atrativos durante o processo de aprendizagem.

Pesquisadores da área estão tentando tornar as disciplinas de programação mais lúdicas e mais próximas à realidade dos estudantes. Nos últimos anos, a gamificação está sendo amplamente aplicada na fase de construção de algoritmos, depuração e refinamento de softwares (Portela, 2020) (Venter, 2020) (Combéfis et al., 2016).

Zapata (Zapata et al., 2020), criaram um modelo de gamificação durante o ensino da disciplina de Algoritmos e Programação na instituição pública Politécnico Colombiano Jaime Isaza Cadavid, Colômbia. O modelo dos pesquisadores premiava os alunos com pontos e com medalhas à medida que eles iam resolvendo os exercícios práticos. Ao final do projeto, os autores citaram a melhoria de desempenho dos estudantes, uma participação mais ativa na disciplina, um maior desempenho durante o curso e um aumento na capacidade de trabalho em equipe.

Fuchs e Wolff (2016) propuseram um jogo online, Classgame, e o aplicou as suas turmas da University of Regensburg, Alemanha. Centrado no *feedback* formativo e em metas, o jogo consiste em resolver desafios agrupados por nível de dificuldade. À medida que o aluno vai resolvendo os exercícios, o sistema atribui estrelas como *feedback* para a solução proposta. Além do sistema de estrelas, há classificação por pontos de experiência. Novos desafios são liberados de acordo com a pontuação de cada aluno. A ferramenta está em avaliação e os resultados serão publicados no futuro.

Maskelunas (Maskeliūnas et al., 2020) criaram um jogo interativo para o ensino de conceitos de algoritmos e os seus tipos. A ideia dos pesquisadores é ilustrar os conceitos da disciplina através de fases de jogos. A cada fase, um conceito é abordado e o aluno é provocado a resolver o desafio do jogo. Ao final da pesquisa, foi demonstrado um melhor desempenho dos alunos que utilizaram o jogo como complemento no ensino.

Carreno (Carreno-Leon et al., 2018) utilizaram técnicas de gamificação na disciplina de introdução à programação da University of Baja California Sur, México. Inicialmente, utilizando *post-its*, o professor indicou todos os passos para a resolução de um algoritmo. Em grupos de quatro alunos, eles deveriam pôr em sequência correta os *cards* responsáveis para resolver o problema. Na fase dois, mais *cards* foram dados aos alunos e, além de colocar os *cards* em ordem, os alunos deveriam selecionar os corretos. Para mensurar os resultados, foram aplicados testes aos grupos e foi verificada a quantidade de exercícios resolvidos por cada um. Após as rodadas de jogos, ficou evidente o crescimento do poder de resolutividade de todos os alunos participantes.

3. Metodologia

Esta etapa foi desenvolvida a partir dos moldes de revisões usadas por Levac (Levac, Colquhoun, O'Brien, 2010), visto que, em seu trabalho, existiam cinco etapas importantes para desenvolver os métodos da pesquisa. No presente artigo, foram utilizadas tais fases, que são: identificar de que se trata a pesquisa; evidenciar os estudos importantes; escolher tais estudos; mapear os dados; e a parte de reunir, sumarizar e relatar os dados. Desta forma, o processo de desenvolvimento dos jogos foi dividido em quatro etapas: revisão da literatura, definição do escopo, implementação da versão em tabuleiro e implementação da versão digital. As quais estão detalhadas nos próximos tópicos.

3.1 Revisões da Literatura

O objetivo principal desta revisão é verificar a viabilidade da utilização de jogos no auxílio ao aprendizado dos alunos de informática em programação. Para um melhor direcionamento, a revisão foi centrada em duas perguntas de pesquisa: quais as técnicas foram usadas para desenvolver as gamificações e quais os resultados das pesquisas. Feito isso, foram criadas as perguntas de pesquisa derivadas, são elas:

1. **A utilização de jogos facilita o aprendizado de programação?**
2. **Há melhora no engajamento dos alunos que utilizam jogos como ferramenta de ensino?**
3. **As taxas de reprovação e de abandono de curso são melhoradas com a utilização de jogos?**

Após a definição da questão primária da pesquisa e suas derivações, o próximo passo foi definir quais bases de dados seriam relevantes para a revisão. Para o levantamento do estudo, foram utilizadas as bases eletrônicas Periódico Capes e Google Scholar.

Logo em seguida da definição das bases, foram estabelecidos quais termos formariam as *strings* de busca. Esses termos fazem referência à aprendizagem, à programação, aos jogos e ao ensino. As melhores combinações de *strings* foram definidas a partir de testes. A cada rodada, alguns ajustes foram feitos e as seguintes *strings* foram definidas como finais:

- Bases Google Scholar: (“*programação*”) AND (“*ensino or aprendizagem*”) AND (“*jogos*”);
- Bases Periódico Capes: (“*jogos*”) AND (“*programação*”) AND (“*ensino or aprendizagem*”);

Como a área da programação de jogos e de gamificações é muito ampla e está em constante evolução, foram definidos alguns critérios de inclusão e exclusão para facilitar a pesquisa. São eles:

- INC01 - Utilização de técnicas de gamificação para auxiliar no estudo
- INC02 - Ensino sobre programação ou área da ciência da computação
- INC03 - Resultados sobre a aplicação da gamificação e a utilização dos jogos
- INC04 - Artigos em português, inglês ou espanhol.
- EXC01 - Publicações duplicadas;
- EXC02 - Publicações não enquadradas nos critérios de inclusão.

Após estabelecidos os critérios, a pesquisa foi iniciada. Durante a seleção dos estudos, as seguintes atividades foram realizadas: 1 – eliminar os artigos duplicados, 2 – leitura rápida do título e resumos 3 – aplicar os critérios de inclusão e de

exclusão definidos. Os artigos que não atendiam aos critérios de inclusão foram descartados. Os artigos com opiniões conflitantes foram submetidos.

A equipe definiu quais informações iriam ser extraídas dos artigos selecionados, que foram: dados bibliográficos, quais técnicas e jogos foram utilizados, qual país o estudo foi realizado e resultados das pesquisas. Adicionalmente, os revisores acrescentaram as suas considerações em relação aos artigos. Por fim, as informações coletadas foram organizadas através de planilhas eletrônicas.

3.2 Definição do Escopo

A segunda etapa consistiu na definição dos temas da disciplina de Algoritmo e Lógica de Programação que seriam utilizados no desenvolvimento dos jogos. Uma pesquisa foi desenvolvida para o levantamento dos principais temas que os alunos apresentam maior dificuldade. Os temas utilizados nos jogos foram escolhidos com base nos resultados desse *survey*. A pesquisa foi destinada aos alunos de Algoritmos e Lógica de Programação. Nela, foi questionado se o primeiro contato que os estudantes tiveram com programação foi através da disciplina; qual dispositivo o aluno utilizava para estudar; quanto tempo, semanalmente, o aluno se dedicava para o estudo de programação; o nível de conhecimento em matemática, de interpretação de texto e de mapear o problema proposto nas questões de Algoritmos e Lógica de Programação, bem como os assuntos que eles tiveram mais facilidade e dificuldade dessa matéria; o grau de dificuldade desse tema e os motivos que dificultaram o aprendizado.

3.3 Implementação da Versão em Tabuleiro

Nesta etapa, foram definidas as regras dos jogos e os materiais que seriam utilizados para o desenho e para a diagramação dos jogos. A plataforma escolhida para essa tarefa foi o site draw.io que é um site de criação de diagramas, o qual é possível utilizar formas geométricas e imagens para fazer os modelos dos jogos.

Para cada jogo foram descritas as regras e as etapas necessárias para concluí-los. Para facilitar a atividade, os jogos foram subdivididos em fases. O jogo da programação foi subdividido em três partes: entender o problema, criar as variáveis e definir os comandos de decisão, já o refinamento foi fracionado em refinamento lógico e refinamento de recursos. A fase de testes, por não ser necessária essa repartição, foi mantida em fase única.

3.4 Implementação da Versão Digital

A implementação digital foi feita por meio de uma plataforma digital de desenvolvimento de jogos. Essa etapa foi dividida em quatro partes, que serão descritas com mais detalhes à frente.

Em primeiro lugar, foi definido o que seria feito para implementar a versão digital do tabuleiro. Com base na precisão e nos recursos necessários para atingir os objetivos requeridos, foi selecionada a Flippity que é uma ferramenta que adapta atividades, transformando-as em jogos para melhorar a atenção e o desempenho dos alunos. Ela funciona da seguinte forma: ao ler o manual de instruções do jogo escolhido, faz-se uma cópia de uma planilha eletrônica de *template* e, com base nela, é possível editar e criar a programação do jogo.

Na segunda parte, as interfaces gráficas dos três jogos foram confeccionadas. Os elementos inseridos nos jogos (as figuras nos escaques contendo o passo a passo a ser seguido e os *templates*), foram editados utilizando o Canva, uma plataforma de design gráfico disponível online e em aplicativos. Após a finalização dos elementos no Canva, devido a uma limitação do Flippity, as imagens só poderiam ser inseridas na ferramenta, caso o upload delas fosse feito anteriormente no Google desenhos, para, enfim, o link específico de cada imagem ser adicionado na Tabela.

Em um terceiro momento, os testes unitários e os sistêmicos foram realizados. Uma planilha de testes foi produzida a fim de detectar possíveis falhas nos jogos, com isso, um relatório contendo os cenários de testes e seus respectivos resultados foi feito. Por fim, após a realização dos testes, os jogos foram publicados e disponibilizados para a utilização.

4. Resultados e Discussões

A revisão da literatura foi feita de acordo com o protocolo da revisão sistemática definida. Houve a seleção dos artigos mais relevantes que estavam dentro dos critérios de inclusão e exclusão e, após a leitura de títulos e de resumos, seis artigos foram aprovados para se tornarem a base de estudo deste projeto.

Quadro 1 - Levantamento dos países, do período, dos dados utilizados e das suas origens para os trabalhos selecionados.

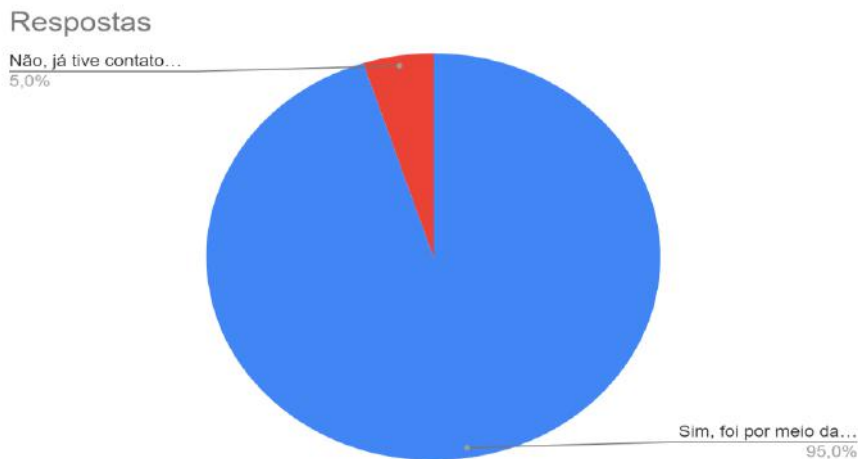
Título	Bibliografia	País de teste	Técnicas e jogos	Resultados	Considerações
Aprendizagem e Jogos: diálogo com alunos do ensino médio-técnico	Amorim, M. C. M. D. S., Oliveira, E. S. G., Santos, J. A. F., & Quadros, J. R. D. T. (2016). Aprendizagem e Jogos: diálogo com alunos do ensino médio-técnico. <i>Educação & Realidade</i> , 41, 91-115.	Brasil	Foi utilizada a linguagem Java, e a interface gráfica do usuário (GUI - Graphical User Interface). Jogo da memória, Candy Crush, Bejeweled, jogo da velha, senha	Maior interesse dos alunos em aprender o conteúdo e participar das aulas.	Aborda a teoria de como jogos podem ajudar no aprendizado, e dá alguns exemplos
Estratégias Pedagógicas No Ensino De Algoritmos e Programação Associadas Ao Uso De Jogos Educacionais	Rapkiewicz, C. E., Falkembach, G., Seixas, L., dos Santos Rosa, N., da Cunha, V. V., & Klemann, M. (2006). Estratégias pedagógicas no ensino de algoritmos e programação associadas ao uso de jogos educacionais. <i>RENTE</i> , 4(2).	Brasil	O jogo que foi desenvolvido: Jogo MaGu	Maiores índices de aprendizagem	Mostra estratégias para o uso de jogos para a aprendizagem de programação.
Integrando Jogos de Lógica Matemática no Ensino de Algoritmos: Relatos de Experimentos	de Souza, M. B., & Moreira, J. L. G. (2015). Integrando Jogos de Lógica Matemática no Ensino de Algoritmos: Relatos de Experimentos. <i>RCT-Revista de Ciência e Tecnologia</i> , 1(1).	Brasil	IDE 's Free Pascal (FP) e Geany, Jogo da Velha (JV) e das Tabelas Mágicas.	Melhores percentuais de aprovação. Redução no total de estudantes que nunca apareceram para assistir às aulas. Porém, houve um aumento no número de Melhores percentuais de aprovação. Redução no total de estudantes que nunca apareceram para assistir às aulas. Porém, houve um aumento no número de reprovados.	Utilizando técnicas de programação, desenvolveu nos alunos aptidão para obterem sucesso no desenvolvimento de algoritmos.

Atraindo Alunos do Ensino Médio para a Computação: Uma Experiência Prática de Introdução à Programação utilizando Jogos e Python	Marques, D. L., Costa, L. F. S., de Azevedo Silva, M. A., & Rebouças, A. D. D. S. (2011, November). Atraindo alunos do ensino médio para a computação: Uma Experiência Prática de Introdução à Programação utilizando Jogos e Python. In Anais do Workshop de Informática na Escola (Vol. 1, No. 1, pp. 1138-1147).	Brasil	Utilizando Python e a biblioteca Pygame foram desenvolvidos alguns jogos simples, como o desenvolvimento de um jogo da velha com interface gráfica.	Aumento do desempenho dos alunos nas avaliações relacionadas ao ensino de lógica de programação, aumento da motivação do aluno na disciplina, no conhecimento e habilidades na área de lógica de programação.	Apresenta uma oficina cujo objetivo foi ensinar noções de programação para alunos do ensino médio, através de jogos digitais, utilizando Python, para motivar os alunos, aumentar seu interesse para a área de programação, etc.
Jogos com Propósito para o Ensino de Programação	Monclar, R. S., Silva, M. A., & Xexéo, G. (2018). Jogos com Propósito para o Ensino de Programação. Anais do XVII Simpósio Brasileiro de Jogos e Entretenimento Digital-SBGames, 1132-1140.	Brasil	Robocode, RoboZZle, Codecombat, Robo Logic,	iniciativas educacionais para o desenvolvimento ou uso de jogos na programação para ensinar um conhecimento que se mostra relevante, porém, atualmente, desestimulante.	Possui um foco introdutório para criar os alicerces computacionais que posteriormente podem ser utilizados em disciplinas acadêmicas em programação.
Ensino de programação utilizando jogos digitais: uma revisão sistemática da literatura	Medeiros, T. J., da Silva, T. R., & da Silva Aranha, E. H. (2013). Ensino de programação utilizando jogos digitais: uma revisão sistemática da literatura. RENOTE, 11(3).	Brasil	PyGame, RoboMind, Lego Mindstorms, Takkou, Scratch, Alice, iVprog, Escracho, Kodu, Game Maker e Construct 2	Os resultados obtidos mostraram que as pesquisas focam principalmente no ensino de programação para o ensino médio e a maioria está sendo realizada por pesquisadores localizados nas em três regiões brasileira	Tem um objetivo bem explícito: as revisões sistemáticas, gamificação, e a utilização de jogos para o ensino da programação.

Fonte: Autores (2021).

Em seguida foi criada uma pesquisa com o objetivo de identificar quais foram os temas e as maiores dificuldades dos alunos na disciplina de Algoritmos e Lógica de Programação; 57% dos alunos responderam o formulário de pesquisa. A sumarização das respostas para a pergunta sobre quando foi o primeiro contato dos alunos com a programação está presente no Gráfico 1.

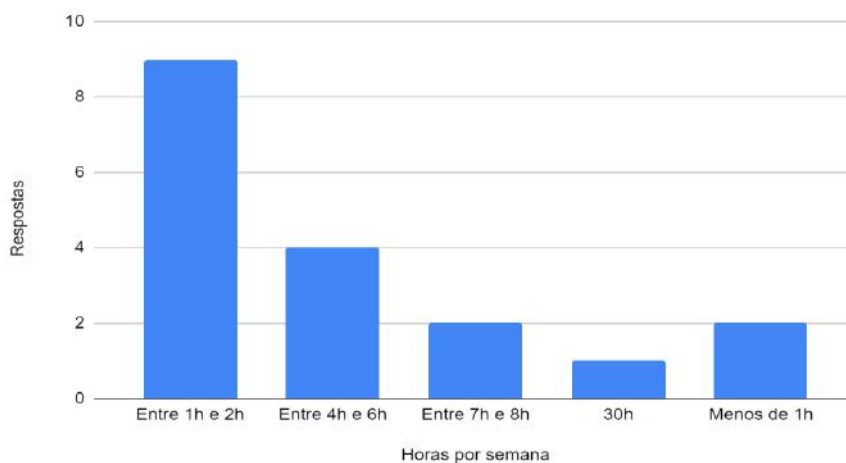
Gráfico 1 - Primeiro contato com programação.



Fonte: Autores (2021).

É possível ver no gráfico que majoritariamente todos os alunos do curso de informática tiveram sua primeira experiência com programação nas aulas da disciplina de Algoritmo e Lógica de Programação. O Gráfico 2 contém as respostas referentes à quantidade de horas dedicadas ao estudo da disciplina durante a semana.

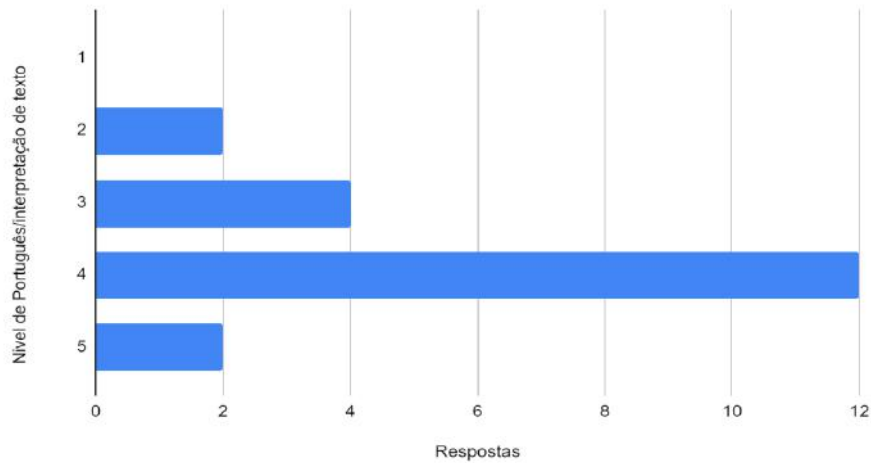
Gráfico 2 - Horas semanais.



Fonte: Respostas obtidas a partir da pesquisa.

A média de horas semanais dedicadas pelos alunos ao estudo da disciplina de Algoritmo e lógica de programação em sua maioria é entre uma e duas horas. Na sequência, o Gráfico 3 ilustra a autoavaliação sobre o domínio da disciplina Português e sobre interpretação de texto.

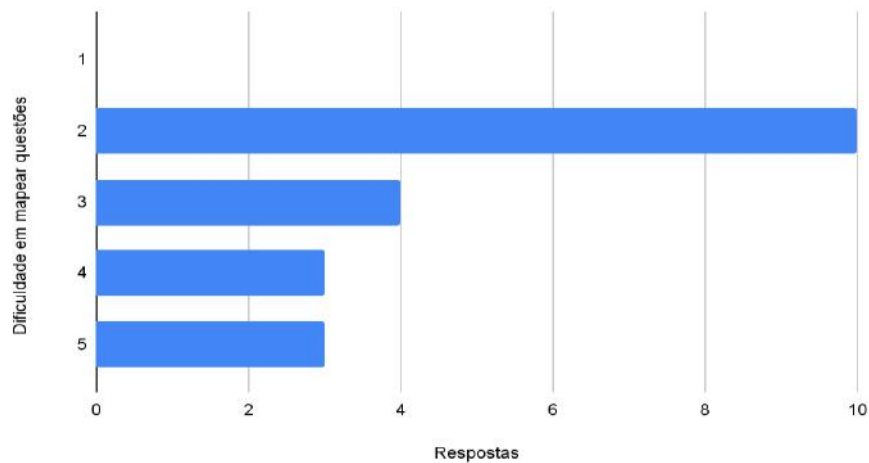
Gráfico 3 - Nível de português e interpretação de texto.



Fonte: Respostas obtidas a partir da pesquisa.

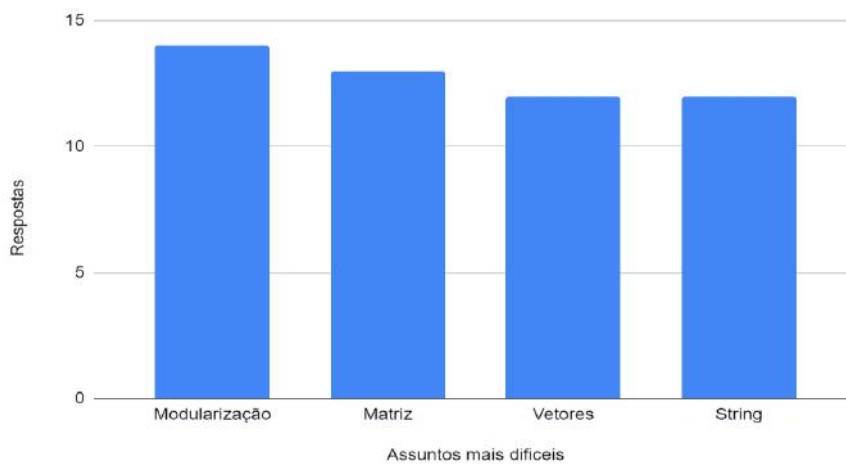
O nível de português e de interpretação de texto foi classificado em uma escala de um a cinco, tendo em vista que os alunos do curso de informática têm um nível considerado alto. Os Gráficos 4 e 5 representam, respectivamente, a dificuldade dos alunos em mapear questões e os assuntos eles julgam ser os mais difíceis na programação.

Gráfico 4 - Dificuldade em mapear questões.



Fonte: Respostas obtidas a partir da pesquisa.

Gráfico 5 - Assuntos mais difíceis na disciplina de Algoritmos e Lógica de Programação.

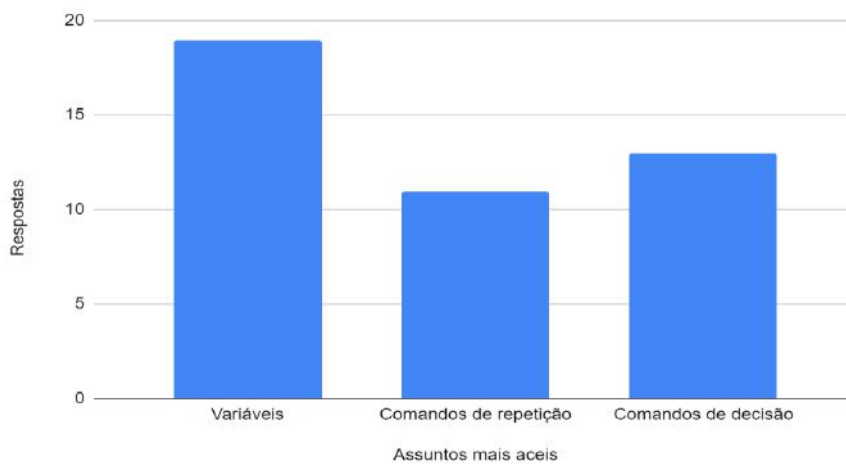


Fonte: Respostas obtidas a partir da pesquisa.

Quanto à dificuldade de mapear questões, a maioria dos alunos declara que apresenta pouca dificuldade ao realizar essa tarefa.

A modularização e a matriz obtiveram mais respostas no quesito de assuntos que os alunos apresentam mais dificuldades. Os Gráficos 6 e 7 demonstram os assuntos que os alunos julgaram mais fáceis de aprender e o julgamento a respeito sobre o grau de dificuldade no aprendizado da matéria.

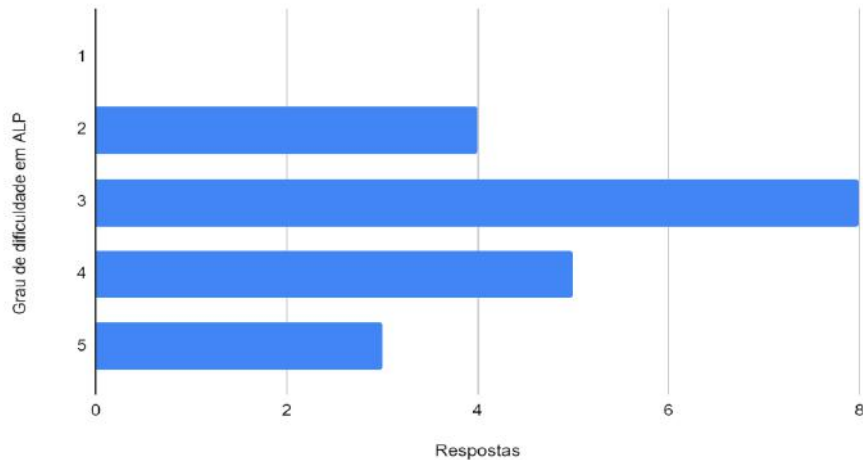
Gráfico 6 - Assuntos mais fáceis.



Fonte: Respostas obtidas a partir da pesquisa.

Os temas que os alunos declararam ser mais fáceis são: Variáveis, Comandos de repetição e comandos de decisão.

Gráfico 7 - Grau de dificuldade na disciplina de Algoritmo e lógica de programação.



Fonte: Respostas obtidas a partir da pesquisa.

E por fim o grau de dificuldade que os alunos tiveram na disciplina de Algoritmo e Lógica de programação está representado no Gráfico acima.

O processo de aprendizagem de programação requer tempo e dedicação tanto de quem está aprendendo quanto de quem ensina, além do fato de que diversos fatores podem influenciar no aprendizado. Os alunos relataram alguns fatores que impactam em seus estudos, dentre as dificuldades citadas, podemos destacar a falta de tempo, a falta de prática na disciplina, a dificuldade no entendimento do enunciado da questão e na adaptação ao ensino remoto ocasionado pela pandemia.

Levando-se em consideração os aspectos apresentados nos itens anteriores, foram formulados três jogos para auxiliar no aprendizado sobre a matéria de Algoritmos e Lógica de Programação. A figura 1 ilustra a versão de tabuleiro do primeiro jogo: o jogo da programação. O objetivo do jogo é resolver um problema proposto através de programação, sendo está dividido em três fases:

Fase 1 – Entendendo o problema.

Os objetivos da fase um são: entender o problema, mapear quais são as entradas e saídas. Adicionalmente, o jogador precisará levantar as regras, as restrições e os cálculos necessários.

Fase 2 – Mapeando em variáveis.

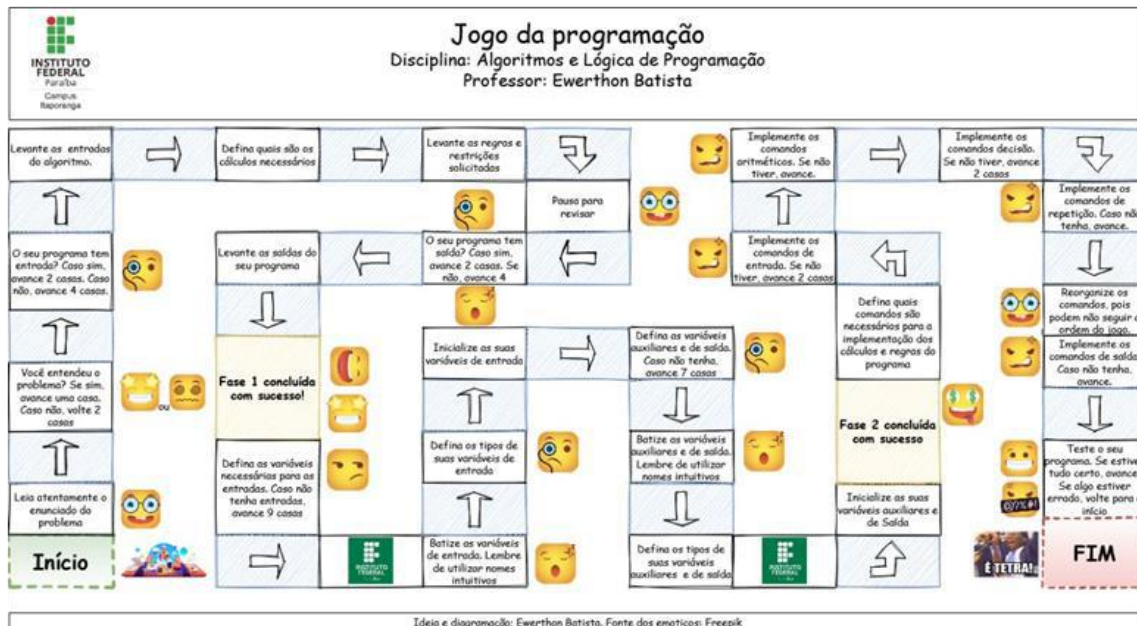
A fase dois objetivou a criação das variáveis. O jogador deve lembrar que é muito importante escolher o tipo correto para cada variável e o nome sugestivo.

Fase 3 – Desafio final.

As variáveis que serão utilizadas devem ser determinadas ou alteradas por cada um dos comandos. Por fim, teste se o seu programa produz o resultado esperado.

Em relação à dinâmica do jogo, cada retângulo é referente a uma ação (passo) necessária para criar programa. O jogador sempre deve obedecer aos segmentos das setas e só avançar caso tenha 100% de certeza que cumpriu a atividade solicitada; as orientações de revisões devem ser respeitadas. Ganha quem chegar ao fim do tabuleiro com o algoritmo rodando e produzindo as saídas solicitada.

Figura 1 - Jogo da programação versão em tabuleiro



Fonte: Autores (2021).

O segundo jogo, representado pela Figura 2, foi desenvolvido para auxiliar os alunos na etapa de refinamento dos seus códigos. O jogo se passa em um ambiente virtual em que o programador irá analisar o seu código e melhorá-lo tanto quanto possível, ou seja, refina-lo e torná-lo mais simples. O jogo é dividido em duas fases:

1. Refinamento de Recursos

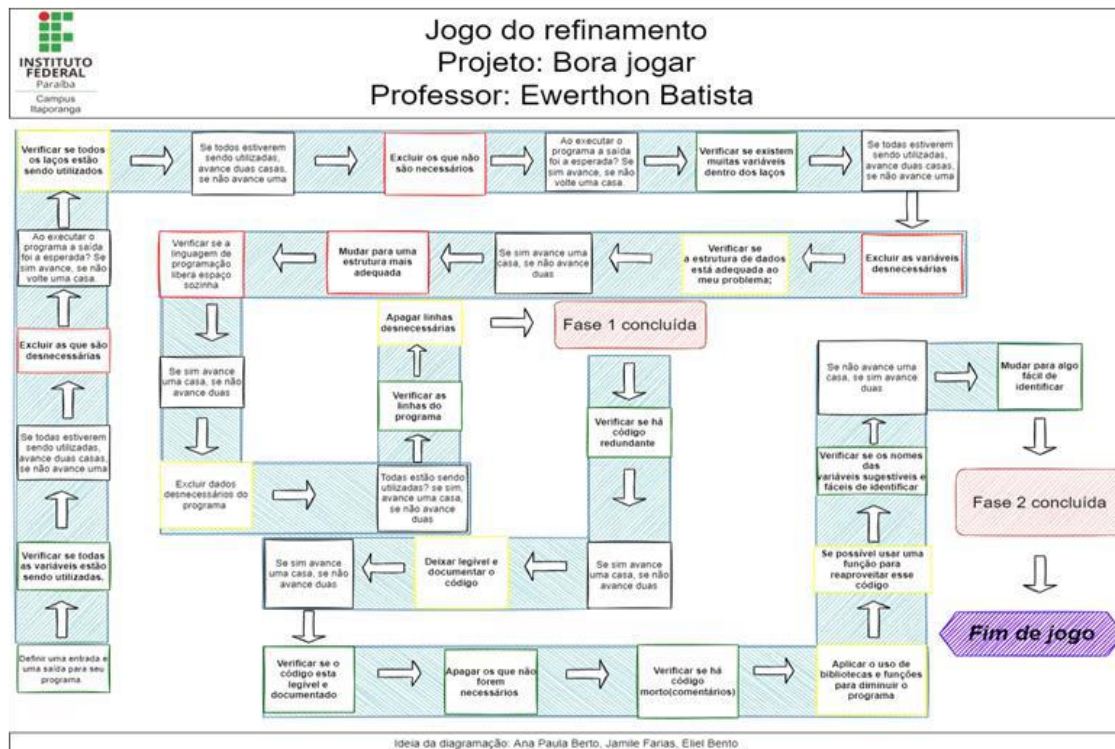
O objetivo do jogador é refinar todos os recursos possíveis de seu código: variáveis inutilizadas serão descartadas, laços que não são necessários, deletados, dentre outras coisas. Apenas siga para a próxima fase se todos os recursos foram devidamente refinados.

2. Refinamento lógico

A parte de refinamentos será finalizada fazendo um refinamento lógico do programa: apagando linhas desnecessárias, reaproveitando código, utilizando bibliotecas e funções prontas para reduzir o código, etc. Lembre-se de rodar seu código mais uma vez e ver se ele está imprimindo a mesma saída com os resultados esperados.

Cada fase é dividida em várias tarefas, uma vez que a tarefa é concluída com sucesso o jogador avança uma casa. As tarefas são baseadas em sua dificuldade: verde (fácil), amarelo (médio) e vermelho (difícil). Se a saída não for a esperada ao final de uma fase o jogador deverá voltar ao ponto de partida. Os jogadores só podem avançar para a próxima tarefa, se a tarefa anterior for concluída corretamente.

Figura 2 - Jogo do refinamento versão em tabuleiro.

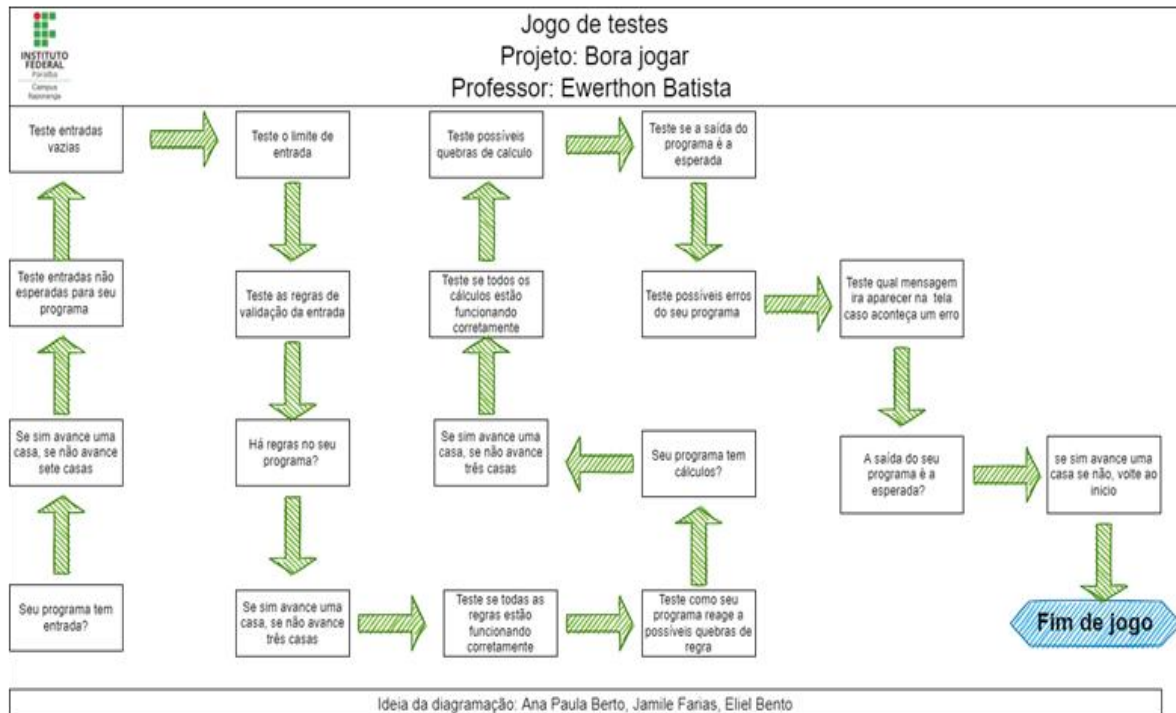


Fonte: Autores (2021).

Segundo Rios e Moreira (2006) uma das partes mais importantes da programação é a fase de testes. É nela que é testado o programa para verificar o que irá ocorrer, se algo inesperado acontecer, por exemplo, uma entrada maior do que a prevista ou um erro de digitação do usuário.

A Figura 3 ilustra o terceiro jogo desenvolvido. Nesse, é realizado diversos tipos de testes para que o jogador faça no seu código. Seu objetivo é testar procedimentos para detectar possíveis falhas. Cada quadro do tabuleiro de jogo equivale a um teste. Após a conclusão do teste, o jogador deve avançar uma casa. Deve-se avançar para o próximo quadro somente quando o teste anterior for concluído com sucesso. Vence quem atingir a linha de chegada com todos os testes realizados com sucesso.

Figura 3 - Jogo de testes versão em tabuleiro.



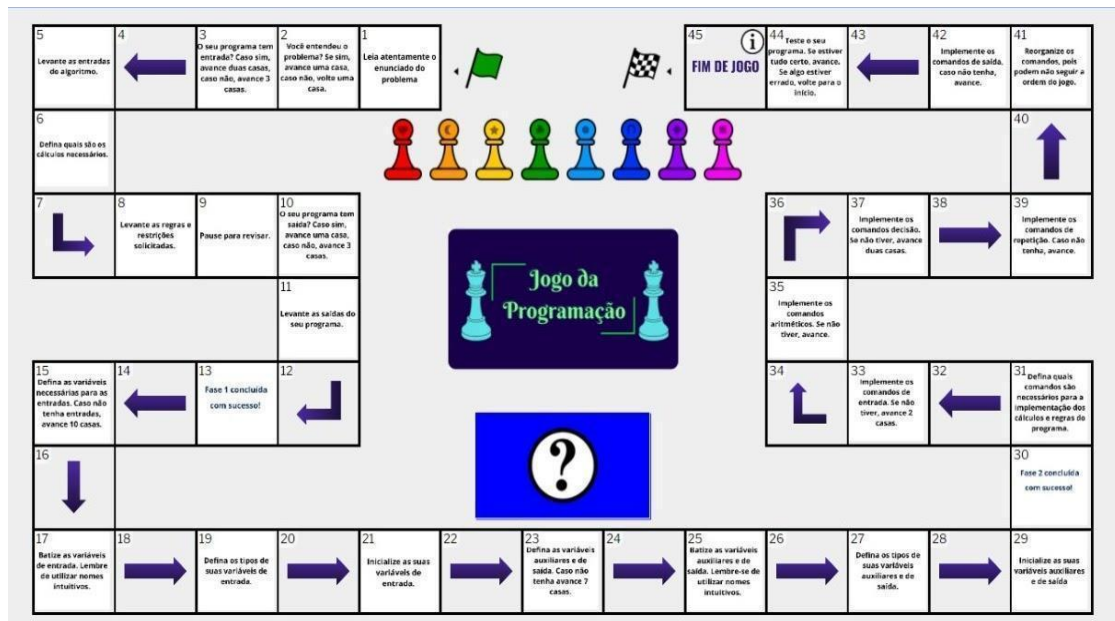
Fonte: Autores (2021).

Superada a fase de desenvolvimento dos jogos de tabuleiro, as versões digitais foram implementadas. Inicialmente, seria usada uma linguagem de programação de alto nível para desenvolver os jogos, porém, com as dificuldades proporcionadas pelo distanciamento social imposto pela pandemia, optou-se por utilizar uma ferramenta menos complexa. Apesar de algumas limitações, o resultado esperado foi alcançado utilizando o site flippity.com.

Após algumas mudanças e adaptações na estrutura dos jogos, como, as alterações do número das casas, a mudança de algumas regras, a etapa do desenvolvimento digital foi concluída.

O *framework* Flippity permite que a programação seja feita a partir de planilhas do Google, onde é possível organizar e editar cada uma das peças dos jogos. As imagens foram feitas na ferramenta Google desenhos, publicadas e linkadas nas planilhas. Em relação ao *template* das figuras, foi utilizado a plataforma Canva. A Figura 4 demonstra a versão digital do jogo da programação.

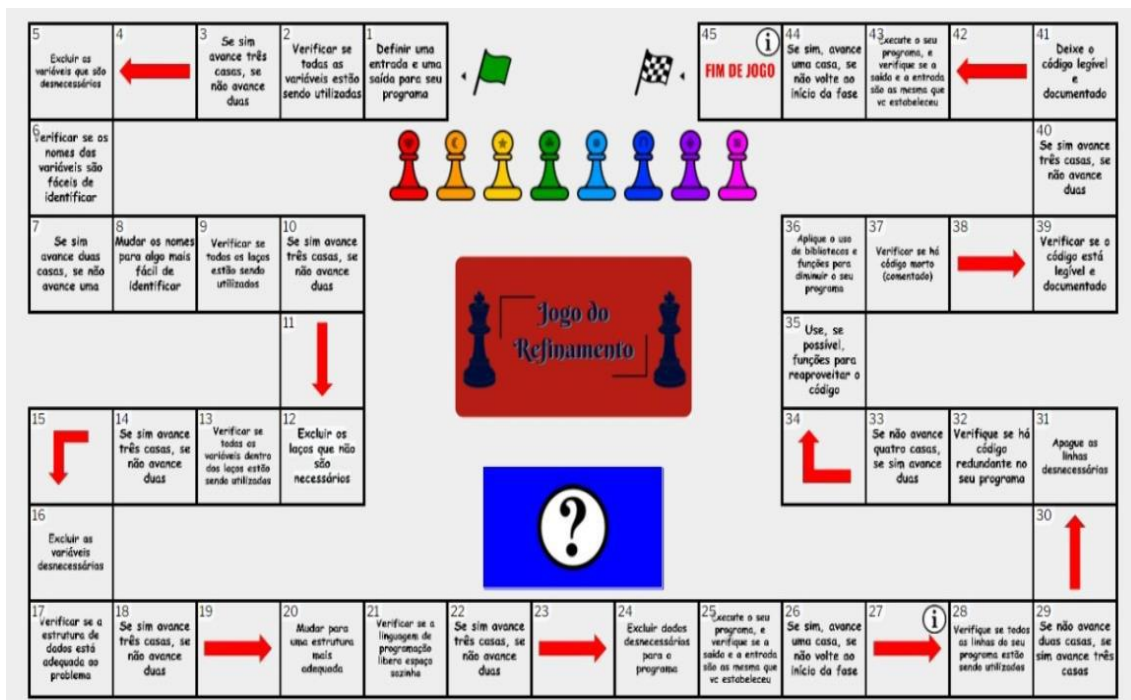
Figura 4 - Jogo da programação versão digital.



Fonte: Autores (2021).

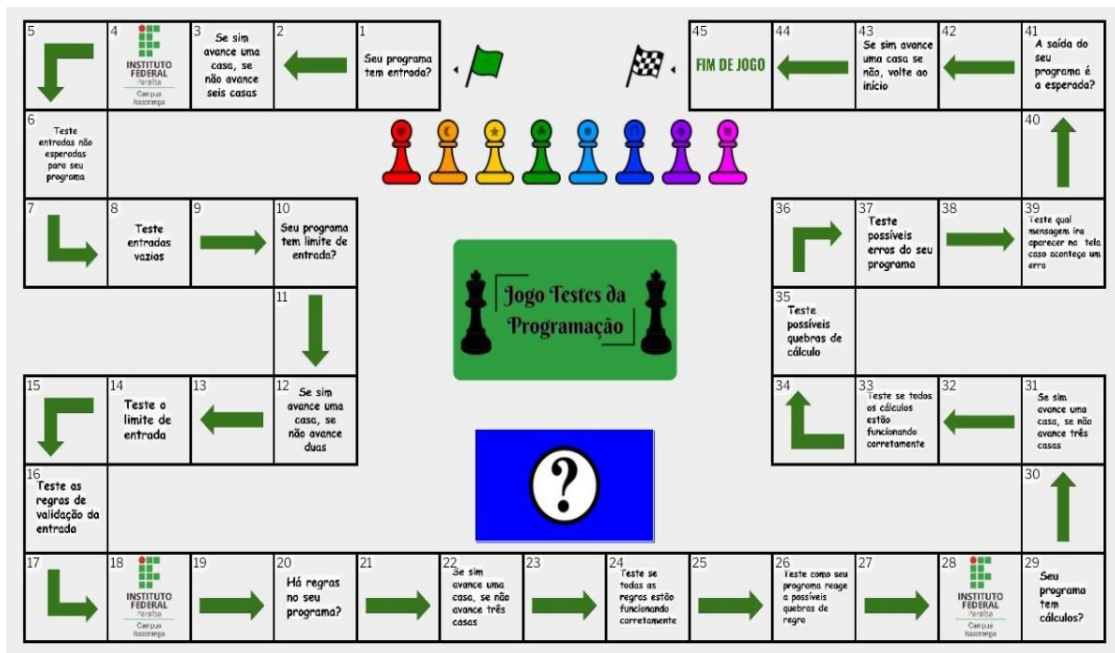
A Figura 5 e figura 6 descrevem, respectivamente, os jogos do refinamento e de testes em suas versões digitais.

Figura 5 - Jogo do refinamento versão digital.



Fonte: Autores (2021).

Figura 6 - Jogo de testes versão digital.



Fonte: Autores (2021).

Nas versões digitais o usuário arrasta a sua peça ao decorrer que avança nas atividades dos jogos. Além disso, ao clicar nas interrogações os jogadores terão acesso a dúvidas e esclarecimentos sobre os assuntos da matéria de Algoritmos e lógica de programação.

Os três jogos digitais estão publicados e disponíveis para o uso dos alunos por meio de browsers de computadores e de notebooks. Devido a algumas limitações da ferramenta Flippity, infelizmente, no celular, o zoom fica impossibilitado. Com isso, ainda não há versões homologadas para o uso através desses dispositivos.

5. Considerações Finais

Observando o cenário da ambientalização no estudo da programação, é perceptível que os alunos sintam um pouco de receio e dificuldade no aprendizado, já que se trata de uma disciplina que envolve tanto raciocínio lógico, como também a prática perante os algoritmos. Dessa forma, a criação dos jogos com essa temática e contendo orientações corretas para a construção de um algoritmo preciso podem auxiliar no desempenho prontamente para o entendimento do estudante de programação, gerando um potencial redutor da taxa de desistência do estudo de programação, que ainda se faz presente na área da tecnologia da informação.

Foram criados três jogos de tabuleiro e digitais, usando a ferramenta draw.io e a Flippity (ferramenta utilizada para criar diferentes tipos de jogos), visando auxiliar os estudantes para que estes tenham um aprendizado mais proveitoso e possam estudar com mais ânimo e êxito. Os jogos servem para criar, refinar e testar algoritmos. Cada jogo possui regras e etapas específicas e o jogador precisa vencer todas as fases para concluir o jogo, atentando sempre às instruções do tabuleiro e não avançar sem ter concluído os passos anteriores.

Portanto, tendo-se elaborado três jogos tanto na versão tabuleiro quanto na versão digital, é esperado o aumento do empenho juntamente com o aprendizado do estudante na disciplina de Algoritmos e Lógica de Programação, proporcionando mais uma dinâmica de estudo. Sabendo-se também que, cada jogo é constituído por uma respectiva fase e contendo suas regras, para que desta forma, o estudante seja direcionado ao seu objetivo final, ou seja, a criação de seu algoritmo da forma correta e que obtenha sucesso.

Sobre as limitações deste estudo, destaca-se o uso dos jogos apenas via computadores e notebooks, pois, como é sabido, infelizmente, ainda temos alunos nos cursos de TIC que não possuem acesso a tais ferramentas.

Para trabalhos futuros fica sugerido a implementação da versão mobile, bem como a criação de novos jogos para abordar novos temas e também continuar auxiliando os discentes na respectiva disciplina de programação. Por fim, após o retorno das aulas 100% presenciais, os jogos produzidos serão inseridos em atividades de ensino da disciplina de Algoritmos e lógica de programação com intuito de mensurar os ganhos alcançados com as utilizações.

Agradecimentos

O desenvolvimento deste artigo contou com o auxílio de algumas pessoas, dentre as quais agradecemos: ao IFPB por todo o apoio financeiro destinado à pesquisa; ao nosso professor orientador, que nos acompanhou, incentivou e ajudou na elaboração deste projeto de pesquisa e aos professores coordenadores que também nos auxiliaram; e, a todos que contribuíram com as pesquisas e colaboraram com a obtenção dos dados utilizados.

Referências

- Amorim, M. C. M. D. S., Oliveira, E. S. G., Santos, J. A. F., & Quadros, J. R. D. T. (2016). Aprendizagem e Jogos: diálogo com alunos do ensino médio-técnico. *Educação & Realidade*, 41, (pp. 91-115).
- Carreño-León, M., Sandoval-Bringas, A., Álvarez-Rodríguez, F., & Camacho-González, Y. (2018, April). Gamification technique for teaching programming. In *2018 IEEE Global Engineering Education Conference (EDUCON)* (pp. 2009-2014). IEEE.
- Combéfis, S., Beresnevičius, G., & Dagienė, V. (2016). Learning programming through games and contests: overview, characterisation and discussion. *Olympiads in Informatics* 10(1), (pp.39-60)
- de Carvalho, J. V., Renner Filho, A., de Oliveira, A. C. R., da Silva, V., & da Graça Gomes, M. (2018). Dispositivos vestíveis aplicados no ensino. *Revista Observatório*, 4(3)(pp. 509-539.)
- de Souza, M. B., & Moreira, J. L. G. (2015). Integrando Jogos de Lógica Matemática no Ensino de Algoritmos: Relatos de Experimentos. *RCT-Revista de Ciência e Tecnologia* (p. 1).
- Fardo, M. L. (2013). A gamificação aplicada em ambientes de aprendizagem. *Renote*, 11(1), (pp. 1-11).
- Figueiredo, J., & García-Peñalvo, F. J. (2020, April). Increasing student motivation in computer programming with gamification. In *2020 IEEE Global Engineering Education Conference (EDUCON)* (pp. 997-1000). IEEE.
- Fuchs, M., & Wolff, C. (2016, April). Improving programming education through gameful, formative feedback. In *2016 IEEE Global Engineering Education Conference (EDUCON)* (pp. 860-867). IEEE.
- Gonçalves, T. C., da Silva, J. C., & Cortes, O. A. C. (2018). Técnicas de mineração de dados: um estudo de caso da evasão no ensino superior do Instituto Federal do Maranhão. *Revista Brasileira de Computação Aplicada*, 10(3), (pp. 11-20).
- Guerra, M. S., Neto, H. A., & Oliveira, S. A. (2018). A Case Study of Applying the Classification Task for Students' Performance Prediction. *IEEE Latin America Transactions*, 16(1), (pp. 172-177).
- Hoed, R. M., Ladeira, M., & Leite, L. L. (2018). Influence of algorithmic abstraction and mathematical knowledge on rates of dropout from Computing degree courses. *Journal of the Brazilian Computer Society*, 24(1), (pp. 1-16).
- Jácome-Amores, L., Freire, W. R., & Sánchez, R. S. (2020, June). Gamification as an Educational Strategy to Strengthen Cognitive Abilities of Mathematics in School Children. In *International Conference on Innovation and Research* (pp. 142-150). Springer, Cham.
- Levac, D., Colquhoun, H., & O'Brien, K. K. (2010). Scoping studies: advancing the methodology. *Implementation science*, 5(1), (pp. 1-9).
- Machado, L. D. P., Berkenbrock, C. D. M., Anselmo, G., & Siple, I. Z. (2018). Uma ferramenta colaborativa para apoiar a aprendizagem de programação de computadores. *Revista Brasileira de Computação Aplicada*, 10(1), (pp. 23-29).
- Marques, D. L., Costa, L. F. S., de Azevedo Silva, M. A., & Rebouças, A. D. D. S. (2011, November). Atraindo alunos do ensino médio para a computação: Uma Experiência Prática de Introdução à Programação utilizando Jogos e Python. In *Anais do Workshop de Informática na Escola* (Vol. 1, No. 1, pp. 1138-1147).
- Maskeliūnas, R., Kulikajevas, A., Blažauskas, T., Damaševičius, R., & 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), (p. 102).
- Medeiros, T. J., da Silva, T. R., & da Silva Aranha, E. H. (2013). Ensino de programação utilizando jogos digitais: uma revisão sistemática da literatura. *RENOTE*, 11(3) (pp. 1-7).

- Monclar, R. S., Silva, M. A., & Xexéo, G. (2018). Jogos com Propósito para o Ensino de Programação. *Anais do XVII Simpósio Brasileiro de Jogos e Entretenimento Digital-SBGames*, (pp. 1132-1140).
- Muñoz, M., Negrón, A. P. P., Mejía, J., Gasca-Hurtado, G. P., Gómez-Alvarez, M. C., & Hernández, L. (2019). Applying gamification elements to build teams for software development. *IET Software*, 13(2), (pp. 99-105).
- Paiva, J. C., Leal, J. P., & Queirós, R. (2020). Fostering programming practice through games. *Information*, 11(11), (p. 498).
- Portela, F. (2020). A new and interactive teaching approach with gamification for motivating students in computer science classrooms. In *First International Computer Programming Education Conference (ICPEC 2020)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik (pp. 1-7).
- Puerta, L. N. Z., & Gómez-Álvarez, M. C. (2020, November). Modelo de evaluación gamificado en cursos de algoritmos y programación. In *2020 39th International Conference of the Chilean Computer Science Society (SCCC)* (pp. 1-8). IEEE.
- Puerta, L. N. Z., & Gómez-Álvarez, M. C. (2020, June). Gamified teaching model to promote learning of algorithms and programming. In *2020 15th Iberian Conference on Information Systems and Technologies (CISTI)* (pp. 1-7). IEEE.
- Rapkiewicz, C. E., Falkembach, G., Seixas, L., dos Santos Rosa, N., da Cunha, V. V., & Klemann, M. (2006). Estratégias pedagógicas no ensino de algoritmos e programação associadas ao uso de jogos educacionais. *RENOTE*, 4(2), (pp. 1-7).
- Rios, E., & Moreira, T. (2006). Teste de software. *Alta Books Editora* (pp. 1-7).
- Salvetti, D.D & Barbosa, L.M (1998). Algoritmos. *Makron Books* (pp. 1-7).
- Venter, M. (2020, April). Gamification in STEM programming courses: State of the art. In *2020 IEEE Global Engineering Education Conference (EDUCON)* (pp. 859-866). IEEE.