

Super Mario Logic: um Jogo Sério para Auxiliar no Processo de Ensino e Aprendizagem de Lógica de Programação

Felipe Schmitt Panegalli – IFFar, PPGTER/UFSM felipe.panegalli@iffarroupilha.edu.br

Giliane Bernardi – PPGTER/UFSM - giliane@inf.ufsm.br

Andre Zanki Cordenonsi – PPGTER/UFSM - andrezc@inf.ufsm.br

Resumo. Este artigo tem como objetivo apresentar o desenvolvimento de um jogo sério denominado Super Mario Logic, voltado para apoiar o ensino e aprendizagem de lógica de programação para estudantes de ensino superior, trazendo desafios que buscam trabalhar as principais estruturas algorítmicas. A metodologia Interad foi utilizada para o desenvolvimento das interfaces educacionais digitais e o jogo foi avaliado pelo método GameFlow. Os resultados iniciais demonstram que os estudantes sentiram-se desafiados, imersos no ambiente virtual e engajados em aplicar os conhecimentos de lógica na resolução dos problemas.

Palavras-chave: Lógica e Programação, Algoritmos, Jogos Sérios

Abstract. This paper presents the development of a serious game called Super Mario Logic, designed to support the logic programming teaching and learning for college students, bringing challenges focused in the main algorithmic structures. The Interad methodology was used to develop the digital educational interfaces and the game was evaluated through the GameFlow method. The initial results demonstrate that students felt challenged, immersed in the virtual environment and engaged in applying logic knowledge in solving problems.

Keywords: Logic Programming, Algorithms, Serious Games

1. Introdução

No âmbito dos cursos de graduação ou técnicos que possuem em sua grade curricular disciplinas de algoritmo e programação, constata-se uma dificuldade na compreensão de aspectos de lógica de programação, pois, para que o aluno atinja um bom desempenho no desenvolvimento de algoritmos, o entendimento e conhecimento da lógica é fundamental (Falkembach e Araújo, 2013). Isso exige grande dedicação por parte do estudante para aquisição dos conhecimentos e da habilidade prática associada, bem como muito esforço por parte dos docentes em busca de estratégias e metodologias diversificadas que auxiliem na efetividade da aprendizagem (Coutinho et al., 2018).

Dessa forma, existindo complexidade em compreender a lógica em si, o estudante pode perder o interesse pela programação ou até mesmo desistir de continuar o curso em que se encontra. Considerando esta problemática, a utilização de ferramentas tecnológicas para facilitar a compreensão e entendimento da lógica de programação, por meio de desafios e práticas, é uma abordagem interessante para minimizar as questões apresentadas.

Acreditando que a aprendizagem de lógica de programação pode ser trabalhada de modo interativo e dinâmico, utilizando as tecnologias como mediadoras do processo, apresenta-se, neste artigo, um jogo sério denominado Super Mario Logic, que apresenta uma relação de problemas relacionados à lógica de programação por meio de desafios envolvendo as estruturas básicas usualmente discutidas em disciplinas de lógica e programação. Santana et al. (2018) destacam que as práticas de ensino exercem forte influência sobre a motivação dos estudantes e, assim, propor melhorias nas experiências

pedagógicas pode aumentar a participação e o engajamento dos estudantes nestas disciplinas. Neste contexto, o objetivo principal deste artigo é apresentar o desenvolvimento e a avaliação do jogo supracitado, buscando fornecer uma alternativa dinâmica, interativa e motivacional.

2. Jogos no Contexto Educacional

De acordo com Santos e Isotani (2018), jogos digitais, usualmente, contemplam atividades envolventes e atraentes, especialmente para os usuários mais jovens, o que pode trazer maior engajamento e motivação dos estudantes, se inseridos em ambientes educacionais.

Huizinga (2007) elenca cinco elementos fundamentais para o sucesso de qualquer jogo: 1) Liberdade; 2) Delimitação; 3) Imprevisibilidade; 4) Regulamentos e normas, e 5) Ficção (ou distância da realidade) e que foram tomados como referência para a construção do jogo aqui apresentado. A Liberdade representa a escolha do jogador de com quem e onde jogar, isto é, a liberdade do usuário em poder escolher com qual personagem ele se identifica e para quais lugares ele pode ir com o personagem, independente do gênero de jogo, seja ele RPG, luta ou corrida, entre outros. O elemento Delimitação diz respeito ao espaço em que o jogo será realizado, delimitando o tema e local em que serão realizadas as partidas. Pode representar ambientes, como uma cidade fictícia ou futurística e, até mesmo, um local existente no espaço real como, por exemplo, uma sala de aula. Já a Imprevisibilidade é uma expectativa gerada automaticamente por uma força maior, sorte ou Inteligência Artificial (IA), em que o jogador acaba recebendo um resultado diferente do que esperava. Outro elemento importante são os Regulamentos e Normas, ou seja, as regras do jogo, que devem existir para que todos tenham as mesmas condições iniciais, onde o sucesso é obtido através do esforço, conhecimento e habilidade (Arruda, 2014). Por fim, o elemento Ficção apresenta a possibilidade de colocar o jogador em um ambiente irreal, ou utilizando equipamentos que seriam impossíveis na vida real.

Jogos Educacionais podem ser referenciados como Jogos Sérios (do inglês *Serious Games*), termo utilizado neste artigo, e que se caracterizam por trazer as características de jogos, porém aplicados a um contexto de aprendizagem, seja em âmbito acadêmico ou profissional, com objetivos extrapolando o entretenimento, mas mantendo o caráter lúdico (Sales et al., 2016). Entre as áreas que têm sido exploradas no desenvolvimento e uso de Jogos Sérios, temos a Computação, buscando apoiar o ensino e aprendizagem de Programação.

Para Raabe et al. (2015, p.1), “o primeiro contato dos estudantes com os conceitos de programação, pode ser determinante na forma como perceberão os desafios e enfrentarão as dificuldades inerentes à aprendizagem de lógica”, destacando, em sua pesquisa, que a motivação e envolvimento trazidos pelos jogos podem trazer resultados benéficos ao colocar os alunos frente à aprendizagem enquanto se divertem. No trabalho dos autores, é realizada uma avaliação de alguns jogos relacionados ao tema, destacando-se o Light Bot 2.01, uma solução comercial, também analisado por este trabalho, que utiliza uma notação de blocos de comando para programar os passos de um robô, abordando conceitos de lógica e de funções.

¹ Light-Bot 2.0 – Criado pela Armor Games no link <http://armorgames.com/play/6061/light-bot-20>

Outros trabalhos correlatos, encontrados no meio acadêmico, foram analisados para fornecer subsídios à construção do jogo apresentado neste artigo, incluindo o jogo *Program Your Robot* (Kazimoglu et al., 2012), similar ao *Light Bot*, que tem como objetivo conduzir um robô até um determinado destino, por meio da seleção de blocos de comandos. Borges et al. (2015) apresentam o *KidCoder*, envolvendo desafios em estilo RPG e *puzzles*, que devem ser solucionados por meio da escrita de códigos-fonte. Semelhante ao trabalho anterior, por se tratar de um protótipo, não discute aplicação e avaliação do mesmo. Por fim, destaca-se o jogo *Process Legend* (Nischida et al., 2017), descrito pelos autores como um jogo lúdico-narrativo, onde um herói precisa resolver desafios de programação, por meio de blocos de instrução, para ajudar seu povo.

O que se observou nos jogos analisados é que, em sua maioria, não foram avaliados com estudantes, para discussão de suas possíveis contribuições. Além disso, não consideraram uma metodologia sistemática para o processo de desenvolvimento, seja de cunho pedagógico ou na área de *Game Design*. Considerando o exposto, as próximas seções apresentam o delineamento metodológico e desenvolvimento do jogo *Super Mario Logic*, concebido para apoiar a aprendizagem das primeiras estruturas associadas à Lógica de Programação.

3. Aspectos Metodológicos

Para o desenvolvimento do jogo proposto, optou-se por utilizar a metodologia Interad (Passos, 2011), desenvolvida para a elaboração de Objetos de Aprendizagem (OA), a qual foi considerada adequada para a elaboração do jogo sério, pois contém etapas para o planejamento e desenvolvimento de *softwares* educacionais com foco no desenvolvimento das interfaces e interações do sistema, características fundamentais em um jogo. Além disso, suas atividades permitem caracterizar adequadamente os elementos que Huizinga (2007) destaca como fundamentais em um jogo. A metodologia abrange 5 fases distintas, desde a concepção do jogo até sua avaliação. São elas:

- *Compreensão*: levantamento inicial das necessidades do objeto de aprendizagem ou, no caso, do jogo desenvolvido;
- *Preparação*: definição e elaboração do material (conteúdo) abordado pelo jogo;
- *Experimentação*: a partir desta fase o OA, neste caso, o jogo passa a ser detalhado, construindo modelos conceituais das interfaces e da arquitetura geral do sistema, bem como diagramas que expressem o fluxo de tarefas;
- *Elaboração*: desenvolvimento dos modelos de navegação e elementos de interação do usuário com o jogo; e, por fim
- *Apresentação*: responsável pelo desenvolvimento e finalização do projeto, no qual a avaliação é executada e resultados são analisados.

Para a avaliação do jogo, foi adotado o método *GameFlow* (Neves et al., 2014), que aborda 8 (oito) categorias de análise da usabilidade e experiência do usuário em jogos: Concentração, Desafio, Habilidade, Controle, Objetivos Claros, Feedback, Imersão e interação Social. No caso desta pesquisa, a categoria Interação Social foi retirada, pois o jogo desenvolvido se classifica como *single player*, não trazendo aspectos de interação em sua mecânica. A avaliação foi realizada com 43 estudantes de disciplinas de lógica e programação dos cursos de Engenharia Civil e Ciência da

Computação de duas universidades federais do RS e seus resultados são discutidos na seção 5.

Com relação às tecnologias escolhidas, foi utilizada a linguagem de programação C# e a *engine* de jogos Unity². A escolha da linguagem de programação C# se deve ao fato de ser flexível, pois utiliza, em sua codificação, os princípios de orientação a objetos; e a Unity, pela facilidade no desenvolvimento de jogos em duas ou três dimensões (2D e 3D), além de utilizar C# como linguagem principal de elaboração de scripts. A próxima seção apresenta o desenvolvimento do jogo.

4. Desenvolvimento do Super Mario Logic

O jogo Super Mario Logic foi pensado tomando como referência o jogo Super Mario³, pela sua popularidade entre os jogadores e fácil reconhecimento do personagem. As ações do jogo proposto são mediadas por meio de comandos, sendo que cada comando é específico como: andar para frente, pular, abaixar, pular no inimigo, subir obstáculos, entre outros. Os comandos são executados por meio de utilização de estruturas de programação, sendo que o estudante (jogador) deverá escolher a(s) mais adequada(s) para cada situação/objetivo. Com relação ao objetivo do jogo, este é avançar de fase até resgatar a princesa (como na história original). Ao final de cada fase, o usuário irá aprender/retomar conceitos básicos discutidos na fase em questão, e a cada fase que vai avançando, novos desafios serão aplicados de forma gradual com as estruturas sequenciais, estruturas de decisão e, por fim, a estrutura de repetição. A seguir, as subseções apresentam todas as fases de modelagem e desenvolvimento do jogo, de acordo com a metodologia Interad, descrita na seção 3.

4.1 Compreensão

Conforme mencionado anteriormente, esta fase objetiva trazer as informações (requisitos) iniciais do jogo, sendo que as principais características são listadas a seguir:

Tema: desenvolvimento de um jogo sério para trabalhar a aprendizagem de lógica de programação nos níveis de estrutura sequencial, estrutura de decisão e estrutura de repetição para que o aluno se sinta mais determinado a alcançar seus objetivos e auxiliando no processo de ensino e aprendizagem.

Público-alvo: estudantes de cursos da área de computação ou afins, que tenham a disciplina de lógica de programação ou algoritmos.

Objetivos Pedagógicos: promover o uso de jogos sérios no processo de ensino e aprendizagem; facilitar o ensino e aprendizagem de lógica de programação no nível sequencial, repetição e decisão; estimular o aluno, de modo que ele perceba que o algoritmo é uma importante ferramenta para resolver problemas; estimular a aprendizagem de uma forma mais ativa, dinâmica e motivadora; e operar com a noção de algoritmos, aplicando-a, de modo apropriado, às situações e problemas apresentados.

² <https://unity.com>

³ Nintendo's Official Home for Mario: <http://mario.nintendo.com/>

Contexto educacional: o jogo sério é voltado para a área da computação e pode ser utilizado de várias formas: durante as aulas, em sala de aula ou em laboratório, ou fora do âmbito das aulas, tanto presencialmente como à distância.

Necessidades do aluno: as disciplinas supracitadas exigem raciocínio lógico e habilidades relacionadas ao pensamento computacional e nem todos os estudantes chegam ao ensino superior com estas habilidades cognitivas. As principais estruturas da programação são basilares para todo o entendimento da disciplina e, por isso, são especificadas dentro do jogo proposto: estrutura sequencial, estrutura de decisão e estrutura de repetição.

4.2 Preparação

A preparação do material foi realizada identificando completamente o conteúdo a ser apresentado no jogo, assim como suas funcionalidades e lista de requisitos. Foram implementados os desenhos de conteúdo e os *storyboards* das telas, como apresentados na Figura 1.

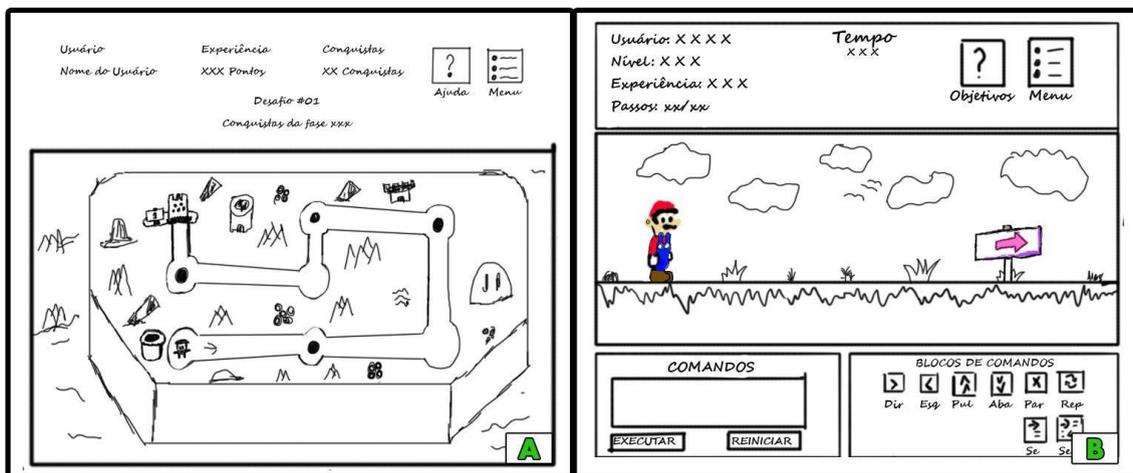


Figura 1 – Storyboards do jogo (Fonte: do autor)

O *storyboard* da Figura 1A apresenta o **mapa**, onde é possível navegar entre os desafios e observar informações como o nome do desafio, conquistas adquiridas, nome de usuário, pontuação, botão de menu e ajuda. Já o segundo *storyboard* (Figura 1B) apresenta a tela **fase**, que é dividida em quatro seções: a) informações do jogo, como nome do usuário; nível; experiência; número de passos para alcançar o(s) objetivo(s) da fase; o tempo necessário para concluir o desafio, entre outros; b) cenário do jogo, onde se encontram o personagem, objetivos e inimigos; c) área da edição dos comandos, seção onde são digitados e executados os comandos, para que o personagem realize as ações dos comandos digitados; e d) blocos de comando, que são blocos com informações (ajuda) de como digitá-los na seção de comando.

4.3 Experimentação

A partir desta fase, o jogo passa a ser detalhado, elaborando o modelo conceitual de interface, o qual é composto por: modelo conceitual, diagrama do MED, arquitetura e fluxo de tarefa. Nesta etapa, foram definidos os signos que irão compor a interface (alguns exemplos podem ser vistos na Figura 2) e uma estrutura de fluxo de informação

linear, através da passagem de fases por parte do jogador. O jogador só alcança as fases subsequentes se conseguir finalizar as fases anteriores, estabelecendo, também, um nível hierárquico de dificuldade.

ELEMENTO	OBJETIVO
	Blocos de comandos, contendo as informações para sua utilização.
	Signo do Mapa, responsável por identificar o usuário.
	Signo do Mapa, responsável por exibir a experiência total do usuário.
	Signo do Mapa, responsável por exibir a quantidade total de conquistas (estrelas adquiridas por fase) do usuário.
	Placa de continuação, ao chegar até a placa o jogo é finalizado.
	Placa de Cuidado, diz ao usuário que um inimigo irá surgir.

Figura 2. Exemplos de alguns signos elaborados para o jogo Super Mario Logic (Fonte: do autor)

4.4 Elaboração

Com o modelo conceitual realizado, a fase da elaboração trata da escolha das interatividades do jogo, onde foi escolhido um desenho de navegação a partir de botões e uma malha estrutural dividida pela disposição dos elementos no leiaute da fase.

4.5 Apresentação

Nesta fase, ocorreu a definição da identidade visual, a conclusão do projeto de interface e o seu desenvolvimento. A Figura 3 apresenta uma coleção das principais interfaces desenvolvidas no jogo, sendo que a Figura 3A apresenta a implementação do *storyboard* disponível na Figura 1A e as funcionalidades já discutidas. A implementação utilizou-se das imagens disponíveis gratuitamente no site da Nintendo⁴. Ao escolher uma das fases disponíveis, o jogador é levado para a interface da Figura 3B. Ao acessar um desafio, inicialmente será exibida uma tela com as informações sobre o mesmo, tais como: o *que* deve ser feito, o *tempo* máximo para cumprir o objetivo e a *quantidade* de passos para chegar até o destino.

Estas são as informações básicas para o aluno cumprir o desafio. Após ler as instruções, ele é conduzido para uma interface, como a da Figura 3C, onde ocorre o jogo propriamente dito, e deve digitar os comandos necessários para cumprir o desafio e apertar o botão *Executar*. O personagem Super Mario irá realizar os passos indicados pelo jogador até o fim. Se eles cumprirem o desafio, ele é encerrado com uma mensagem como a da Figura 3D. Caso contrário, o jogador tem a opção de reiniciar a fase e tentar uma nova combinação.

⁴ Nintendo's Official Home for Mario: <http://mario.nintendo.com/>

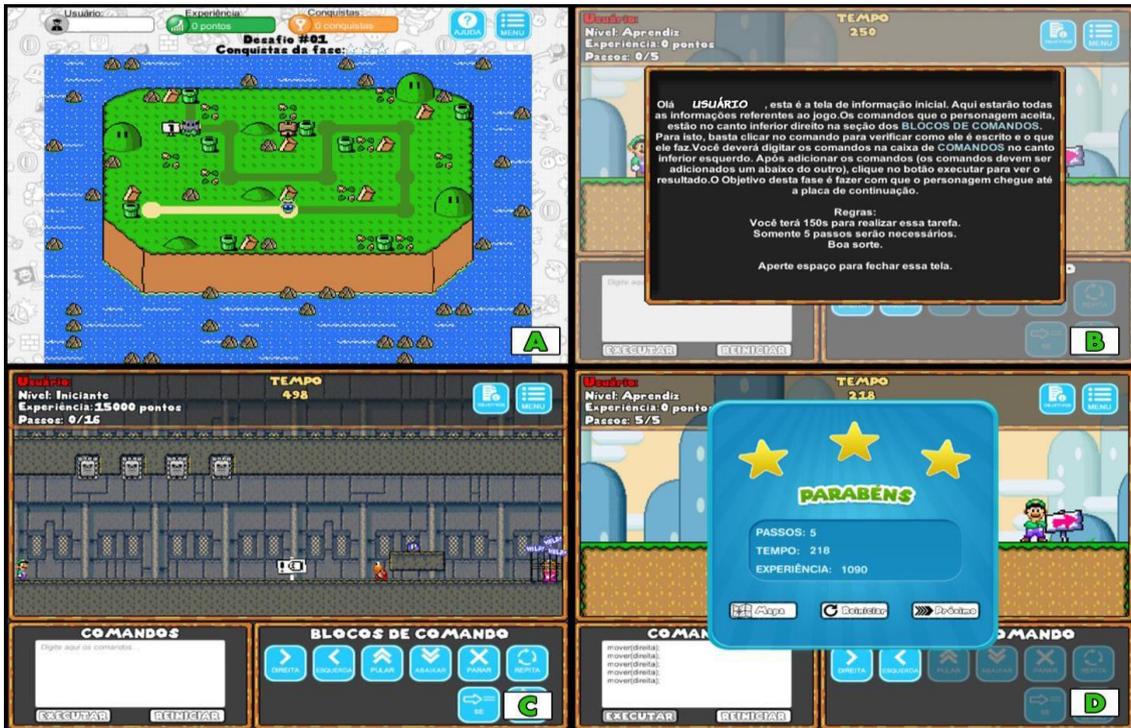


Figura 3. Interfaces do Jogo Super Mario Logic (Fonte: do autor)

5. Avaliação e Discussão dos Resultados

Conforme descrito na seção 3, a avaliação foi realizada com 43 estudantes, por meio do método *GameFlow*, sendo que o instrumento aplicado foi dividido em sete grupos de perguntas entregues em um questionário impresso, cujos resultados estão representados nos gráficos das Figuras 4 a 6. O jogo foi aplicado em dois períodos de 50 minutos.

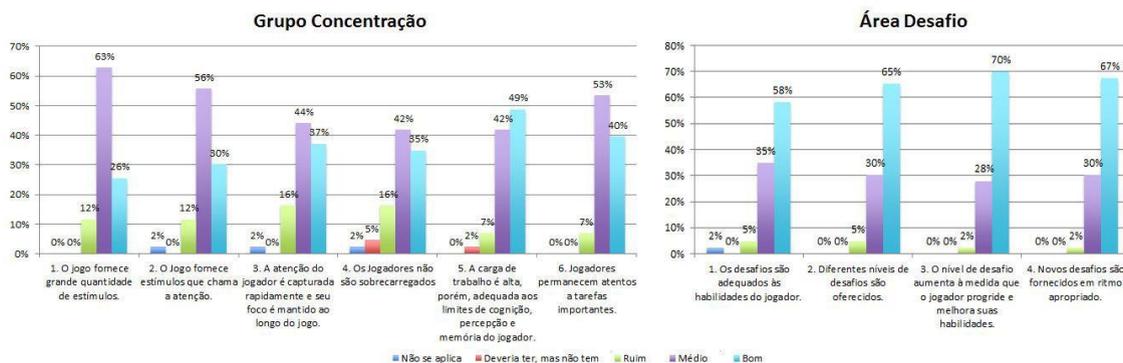


Figura 4. Gráficos para o Grupo Concentração e Área Desafio (Fonte: do autor)

O critério de *Concentração*, de modo geral, obteve um índice médio. O ponto positivo mais alto foi o item 5 que diz “a carga de trabalho é alta, porém, adequada aos limites de cognição, percepção e memória do jogador”, alcançando 91% de aprovação (médio e bom). Este critério vem ao encontro de um elemento de Huizinga (2007), a Liberdade do jogador, ou seja, encontrar maneiras de reter a atenção do usuário com o que interessa para ele, para que consiga jogar de forma adequada e concentrado no que está fazendo. Por outro lado, a avaliação mais baixa foi evidenciada no item 4, que questionava se “os jogadores não são sobrecarregados”, atingindo 16% de avaliações negativas (ruim). Considerando-se o ponto de vista dos estudantes, o que pode ter

contribuído para esta percepção é o número de passos a ser realizado em algumas fases, ocorrendo uma sobrecarga de até oito passos entre um desafio e outro. Os desafios foram pensados de forma que a dificuldade vai aumentando, à medida em que o jogador vai avançando no jogo. Considerando a categoria *Desafio*, apresentada na Figura 4, pode ser observado que houve uma boa avaliação, mostrando que os desafios aplicados no jogo foram bem elaborados e adequados às habilidades dos jogadores.

Na avaliação do grupo de *Habilidades* do Jogador, pode ser visualizado no primeiro gráfico da Figura 5 que o item 01 não obteve um desempenho muito forte. Isto demonstra que o jogo, mesmo sendo intuitivo, não elimina a necessidade de o usuário ler as instruções de como jogar. A cada início de desafio, são mostrados o objetivo e as regras a serem cumpridas, sendo que, caso o usuário não se lembre dos objetivos iniciais, há um botão com tais informações na interface de execução. Isto vem ao encontro do item 3, onde 21 alunos confirmaram a existência de um sistema de Ajuda no ambiente do jogo. A pontuação mais alta foi o do item 5, onde, sob o ponto de vista dos estudantes, o aumento das habilidades do jogador ocorre em um ritmo adequado ao seu progresso no jogo (63% de avaliações positivas). Isto é, o jogo não muda de nível drasticamente do fácil para o difícil e sim de forma gradual para que o usuário progrida conforme as experiências adquiridas nos desafios anteriores.

Ter o poder de realizar ações a partir de tentativas e erros traz ao estudante a sensação de controle que, segundo Neves (2014), é um aspecto importante para levar o jogador a experimentar o estado de controle sobre o(s) elemento(s) do jogo. Essa colocação reflete-se no item 5 do segundo gráfico da Figura 5 (*Controle*), em que 22 alunos responderam ‘bom’ e 17 ‘médio’, ou seja, 91% dos alunos sentem que seus controles e ações são importantes e que refletem no mundo do jogo. Isso vem ao encontro do elemento de Liberdade de Huizinga (2007) sobre a importância de permitir ao jogador tomar suas próprias decisões para alcançar seus objetivos.

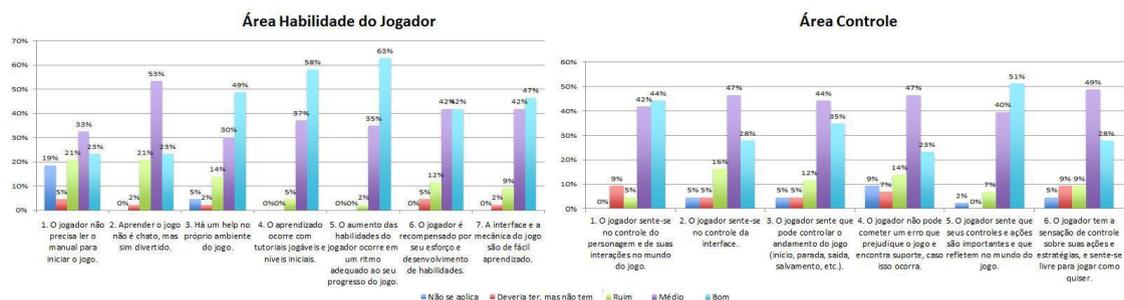


Figura 5. Gráficos para a Área Habilidade do Jogador e Controle (Fonte: do autor)

Ao observar os resultados do grupo *Imersão* de forma geral (Figura 6, gráfico 1), destaca-se que 51% dos estudantes considerou a imersão média (38%) ou boa (13%). Considerando os demais resultados, este pode ser avaliado de forma menos positiva, sendo 19% de respostas na faixa “Ruim” e 6% na faixa “Deveria ter, mas não tem”. No entanto, 24% dos estudantes questionados informaram que as questões de imersão não se aplicavam ao jogo avaliado. Observa-se pelas respostas que, talvez, os estudantes não tenham compreendido de forma adequada o que se desejava com as questões relacionadas com imersão.



Figura 6. Gráficos para a Área Imersão, Objetivos e Feedback (Fonte: do autor)

Quando se trata de objetivos, os mesmos devem ser claros para que o jogador entenda o que deve ser realizado para concluir os desafios. No jogo isso fica evidente no resultado da pesquisa no grupo de *Objetivos* (Figura 6, gráfico 2). Segundo os estudantes questionados, 100% (bom e médio) acreditam que os objetivos principais do jogo (resgatar a princesa) são claros e 91% (bom e médio) acreditam que os objetivos intermediários (objetivos do desafio) também se apresentam de forma clara. Isso mostra que este critério estabelece uma ligação direta com o que é exposto por Salen e Zimmerman (2004) e Huizinga (2007), que destacam que objetivos claros e regras bem definidas fazem com que o jogador consiga se orientar dentro do jogo para realizar o que é solicitado obtendo, assim, uma satisfação pessoal por vencer o desafio aplicando seu esforço, habilidades e conhecimentos adquiridos através dos diversos níveis de dificuldades enfrentados em cada fase. Com relação ao *feedback* do jogo, este se dá em dois momentos. Em tempo real, onde informações são mantidas permanentemente na interface; e informações *estatísticas*, onde o jogador pode obter informações sobre a experiência total adquirida, o nível em que o jogador se encontra, o tempo total de jogo e a porcentagem de conquista por mundo e geral. Este tipo de *feedback* é exibido ao jogador caso ele acesse a opção de Estatísticas, presente na interface. Aplicado o questionário aos alunos, no que se refere ao *Feedback* no jogo, obteve-se uma grande aprovação pelos alunos no item 3 (Figura 6, gráfico 3), “Status e pontuação estão disponíveis ao jogador”, com 95% de respostas do tipo “Bom” e “Médio”. Este resultado pode ser decorrente da disponibilização dos dados em tempo real.

6. Considerações Finais

Este artigo apresentou o desenvolvimento e avaliação do Jogo Sériu Super Mario Logic, voltado para o ensino e aprendizagem de lógica de programação. Ainda, foram abordados os elementos fundamentais para o sucesso de qualquer jogo, conforme mencionado por Huizinga (2007), e que foram necessários para definir os requisitos mínimos a serem utilizados no jogo desenvolvido, tais como a liberdade, em que o jogador pode escolher o personagem com o qual ele mais se identifica, a delimitação, que foi baseada na história original do Super Mario World, a imprevisibilidade, necessária em certos momentos, por meio do surgimento de inimigos (*Thwomps* e *Piranha Plants*), em que o aluno deveria achar o momento certo para não tocá-los e acabar perdendo o jogo. No que diz respeito aos Regulamentos e Normas, estes foram necessários para que todos os jogadores agissem da mesma forma dentro do jogo; e, por fim, a Ficção, pois o jogo se passa no Reino dos Cogumelos, um mundo fictício criado pelo desenvolvedor da franquia Super Mario.

Como sugestões de trabalhos futuros destacam-se: a) mostrar ao final de cada desafio os comandos em outras linguagens de programação como Java, Pascal e C; b)

criar um sistema de *ranking*, onde seja possível visualizar os melhores jogadores para promover o desafio entre os alunos; c) desenvolver um sistema de tempo inicial para cada desafio, fazendo com que o aluno analise o desafio e pense em soluções para o problema proposto.

Por fim, destaca-se que a utilização do jogo Super Mario Logic demonstrou potencial benéfico nos grupos que participaram das avaliações. Os alunos foram desafiados a utilizar o computador como ferramenta, montar uma estratégia lógica para alcançar o desafio primário e secundário, resolver o problema através de abstração de código, implementar a solução com o objetivo de alcançar a combinação de passos e recursos mais eficiente e eficaz e, por fim, transferir o processo de resolução de problemas a uma ampla variedade de problemas através da experiência adquirida em cada problema. Ao se utilizar esta estratégia pedagógica, é possível desmistificar a concepção de que jogos têm como única meta a diversão sem cunho de aprendizagem, podendo facilitar a aprendizagem por meio de diversão e da motivação criada.

7. Referências

- ARRUDA, E. P. **Fundamentos para o Desenvolvimento de Jogos Digitais: S{é}rie Tekne**. [s.l.] Bookman Editora, 2014.
- BORGES, C. et al. KidCoder : Uma Proposta de Ensino de Programação de forma Lúdica. In: **Anais do XXVI SBIE**, v.1, n.2, 2015.
- COUTINHO, E.; BONATES, M.; MOREIRA, L. O. Relato sobre o Uso de uma Ferramenta de Desenvolvimento de Jogos para o Ensino Introdutório de Lógica de programação. In: **Anais dos Workshops do CBIE**. v.7. n.1. 2018.
- FALKEMBACH, G. A. M.; ARAUJO, F. V. DE. Aprendizagem de Algoritmos: Dificuldades na Resolução de Problemas. In: **Anais SULCOMP**, 2013.
- HUIZINGA, J. **Homo ludens: o jogo como elemento da cultura**. 5. ed. São Paulo: Perspectiva, 2007.
- KAZIMOGLU, C. et al. A serious game for developing computational thinking and learning introductory computer programming. In: **Procedia-Social and Behavioral Sciences**, v. 47, 2012.
- NEVES, D. E. et al. Avaliação de jogos sérios casuais usando o método GameFlow. In: **Revista Brasileira de Computação Aplicada**, v.6, n.1, 2014.
- NISHIDA, A.; FERREIRA, V.; BRAGA, J. C. Process Legend: Jogo de Enigmas para o Ensino de Introdução à Programação. In: **Anais dos Workshops do CBIE**. v.6, n.1. 2017.
- PASSOS, P. C. S. J. **Interad: uma metodologia para design de interface de materiais educacionais digitais**. Dissertação (Mestrado) UFRGS, 2011.
- RAABE, A.; ZANCHETT, G.; VAHLICK, A. Jogos de Programar como uma Abordagem para os Primeiros Contatos dos Estudantes com à Programação. In: **Anais dos Workshops do CBIE**. v. 4. n.1, 2015.
- SALEN, K.; ZIMMERMAN, E. **Rules of Play: Game Design Fundamentals**. The MIT Press Cambridge, 2004.
- SALES, A. B.; CLÍMACO, G. S.; SALES M. B. Jogos Sérios em Interação Humano-Computador: Uma revisão Sistemática de Literatura. In: **RENOTE**, v.14, n.1, 2016.
- SANTANA, B.; ARAÚJO, L. G.; BITTENCOURT, R. Considerando a Motivação dos Estudantes em Experiências de Ensino-Aprendizagem de Computação. In: **Anais dos Workshops do CBIE**. v.7. n.1, 2018.
- SANTOS, W. O.; ISOTANI, S. Desenvolvimento de Jogos Educativos? Desafios, Oportunidades e Direcionamentos de Pesquisa. In: **RENOTE**, v.16, n.2, 2018.