



UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO
UNIDADE ACADÊMICA DE SERRA TALHADA
BACHARELADO EM SISTEMAS DE INFORMAÇÃO

JOSÉ ELVIS DA SILVA SOUZA NOGUEIRA

**Desenvolvimento de um Ambiente para
Auxiliar a Prática de Lógica de Programação:
uma abordagem de gamificação**

Serra Talhada,
Março/2021

José Elvis da Silva Souza Nogueira

**Desenvolvimento de um Ambiente para
Auxiliar a Prática de Lógica de Programação:
uma abordagem de gamificação**

Trabalho de Conclusão de Curso apresentada ao Curso de Bacharelado em Sistemas de Informação da Unidade Acadêmica de Serra Talhada da Universidade Federal Rural de Pernambuco como requisito parcial à obtenção do grau de Bacharel.

Orientador: Prof. Dr. Richarlyson Alves D'Emery

Serra Talhada,
Março/2021

Dados Internacionais de Catalogação na Publicação
Universidade Federal Rural de Pernambuco
Sistema Integrado de Bibliotecas
Gerada automaticamente, mediante os dados fornecidos pelo(a) autor(a)

- N778d Nogueira, José Elvis da Silva Souza
Desenvolvimento de um Ambiente para Auxiliar a Prática de Lógica de Programação: uma abordagem de gamificação / José Elvis da Silva Souza Nogueira. - 2021.
30 f. : il.
- Orientador: Richarlyson Alves DEmerly.
Inclui referências.
- Trabalho de Conclusão de Curso (Graduação) - Universidade Federal Rural de Pernambuco, Bacharelado em Sistemas da Informação, Serra Talhada, 2021.
1. Jogos Educacionais. 2. Gamificação. 3. Expressões Lógicas. 4. Avaliação de Usabilidade. I. DEmerly, Richarlyson Alves, orient. II. Título

UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO
UNIDADE ACADÊMICA DE SERRA TALHADA
BACHARELADO EM SISTEMAS DE INFORMAÇÃO

JOSÉ ELVIS DA SILVA SOUZA NOGUEIRA

**Desenvolvimento de um Ambiente para Auxiliar a Prática de Lógica de
Programação: uma abordagem de gamificação**

Trabalho de Conclusão de Curso julgado adequado para obtenção do título de Bacharel em Sistemas de Informação, defendida e aprovada por unanimidade em 03/03/2021 pela banca examinadora.

Banca Examinadora:

Prof. Dr. Richarlyson Alves D'Emery
Orientador
Universidade Federal Rural de Pernambuco

Prof. Me. Carlos André Batista
Universidade Federal Rural de Pernambuco

Prof. Me. Heldon José Oliveira Albuquerque
Universidade Federal Rural de Pernambuco

DEDICATÓRIA

À minha mãe Maria Aparecida, ao meu pai Ernando, meus irmãos Alícia, Eldis, Erlânia e Erlânio por todo suporte e carinho que me deram e continuam a dar. A conquista da conclusão da graduação somente foi possível graças a vocês.

AGRADECIMENTOS

Gostaria de agradecer primeiramente aos meus pais, Maria Aparecida e Ernando, que sempre acreditaram no meu potencial, me apoiando e incentivando a estudar desde muito jovem. Aos meus irmãos por todo o companheirismo e ajuda.

Ao meu orientador Prof. Dr. Richarlyson Alves D'Emery, que sempre esteve disposto a ajudar e orientar, dando um norte neste cenário excepcional em que tivemos que mudar drasticamente a maneira como estudamos.

Aos meus colegas que conheci na Universidade, em especial aos meus amigos João Emerson e Williany Veras que sempre estiveram ao meu lado estudando, fazendo trabalhos acadêmicos e me incentivando a me manter firme nessa estrada.

A todos os professores e funcionários da UAST-UFRPE.

A todos, minha eterna gratidão.

“É necessário sempre acreditar que o sonho é possível, que o céu é o limite e que você truta é imbatível”.

Edi Rock

SUMÁRIO

| | | |
|----------|---|-----------|
| 1 | INTRODUÇÃO | 08 |
| 2 | REFERENCIAL TEÓRICO | 09 |
| 2.1 | Gamificação | 09 |
| 2.2 | Jogos educacionais | 10 |
| 2.3 | Expressões Lógicas | 10 |
| 2.4 | Trabalhos Relacionados | 10 |
| 2.4.1 | Análise Comparativa | 13 |
| 3 | MÉTODO | 14 |
| 4 | RESULTADOS | 19 |
| 4.1 | Boolean Defuse 2.0 | 19 |
| 4.2 | Análise de Dados | 21 |
| 4.3 | Avaliação de Qualidade e Análise e Interpretação por Inferência Estatística | 24 |
| 5 | CONCLUSÃO | 27 |
| | REFERÊNCIAS | 28 |

Desenvolvimento de um Ambiente para Auxiliar a Prática de Lógica de Programação: uma abordagem de gamificação

José Elvis da Silva Souza Nogueira¹, Richarlyson A. D'Emery¹

¹Unidade Acadêmica de Serra Talhada – Universidade Federal Rural de Pernambuco (UAST - UFRPE) - Caixa Postal 063 – 56.909-535 – Serra Talhada – PE – Brasil

Abstract. *Introduction: learning to program is difficult, a fact verified by the high failure rates in courses in the area of Computing, one of the causes being the difficulty of assimilating logical-mathematical concepts. Educational software is an alternative to mitigate this problem. Objective: to present the development of a game that helps in the practice of Logical Expressions. Method: the coding will be in Java. The organization is given in MVC. XML is used for data storage. The game was assessed using the MEEGA methods; SUS and hypothesis testing. Results: it was verified through the SUS evaluation, reconciled the analysis of the applied questionnaire that the game has characteristics that help in learning, in addition to the quality of the same.*

Resumo. *Introdução: aprender a programar é difícil, fato constatado pelas altas taxas de reprovações em cursos de Computação, uma das causas é a dificuldade de assimilação de conceitos lógico-matemáticos. Software educacional é alternativa para mitigar esse problema. Objetivo: desenvolver um jogo que auxilia na prática de Expressões Lógicas. Método: a codificação é em Java e organizada em MVC. Utiliza-se XML para armazenamento de dados. A avaliação é dada por MEEGA, SUS e testes de hipóteses. Resultados: verificou-se através de SUS e análise por inferência estatística a qualidade do jogo e sua contribuição para o aprendizado de Expressões Lógicas.*

1. Introdução

Aprender Programação é uma tarefa de extrema importância aos alunos de cursos de Computação, uma vez que ela é a base para grande parte das disciplinas que o aluno encontrará na sua vida acadêmica. No entanto, isso não é uma tarefa simples, visto que exige, entre outras habilidades, que o aluno consiga entender um conjunto de conceitos abstratos, conhecimentos lógico-matemáticos, apresente capacidade de resolução de problemas, além de entender as regras semânticas de uma linguagem de Programação.

De acordo com [Gomes et al. 2008], existe uma grande taxa de insucesso nas disciplinas introdutórias de Programação, nas quais são ensinados conhecimentos básicos para que o aluno possa resolver problemas reais simples; e que uma porcentagem significativa de alunos apresenta dificuldades em assimilar alguns conceitos abstratos de Programação.

Muitas vezes, os alunos estão habituados a disciplinas em que para receberem boas notas basta que leiam repetidas vezes livros e anotações, se limitando a apenas reproduzir aquilo que foi estudado [Gomes et al. 2008]. No entanto, para programar é preciso mais, é necessário que o aluno faça diversas práticas de modo que consiga compreender o assunto e refletir sobre o mesmo.

As disciplinas que envolvem conteúdos de Algoritmos e Programação são consideradas desafiadoras pelos alunos, pois exige que esses apresentem o desenvolvimento de estratégias para solucionar problemas que possuem base lógico-matemática [Deters et al. 2008]. Como consequência, são ocasionadas diversas dificuldades de aprendizagem nos estudantes, o que propicia a ocorrência de reprovações e desistências.

Sendo assim, o presente artigo apresenta o desenvolvimento do jogo educacional *Boolean Defuse* para auxiliar alunos de cursos de Computação na prática de Expressões Lógicas. A justificativa para a escolha de um jogo digital para auxiliar na aprendizagem se dá pelo fato de o mesmo permitir que o aluno treine a prática, além de, segundo [Alves 2015], ofertar um *feedback* imediato através de características inerentes de *games*, como atingir ou não um objetivo estabelecido, passar de nível, etc. Desse modo, permiti-se que o aluno possa verificar seu desenvolvimento (nível de aprendizagem) e se deve ou não continuar utilizando uma estratégia.

Este artigo está organizado da seguinte maneira: na Seção 2 serão apresentados os referenciais teóricos sobre jogos educacionais, Expressões Lógicas e gamificação, os trabalhos relacionados e uma análise comparativa. Por fim, nas Seções 3, 4 e 5, têm-se o método de pesquisa utilizado, os resultados e a conclusão, respectivamente.

2. Referencial teórico

2.1. Gamificação

Gamificação é um conceito informal definido como o processo de utilização de características dos jogos em um contexto não lúdico, com o intuito de engajar e melhorar a experiência do usuário [Deterding et al. 2011]. De acordo com [Navarro 2013], entende-se como algumas das “características dos jogos” atividades como propor desafios, cumprir regras, metas claras e bem definidas, efeito surpresa, acontecimentos ocorrem de forma linear, conquista de pontos e troféus, estatísticas e gráficos com o acompanhamento de desempenho, superação de níveis e criação de personagem.

Para [Liu et al. 2011], o objetivo principal da gamificação é fazer com que o usuário de um sistema assumo o “comportamento de jogador”, o qual consiste nas capacidades de manter o foco na tarefa que está realizando, executar várias tarefas de forma simultânea sob pressão, trabalhar mais sem descontentamento e sempre tentar novamente quando falhar.

Apesar de não ter sido pensada inicialmente para o ambiente educacional, a gamificação se apresenta como uma alternativa interessante para chamar a atenção dos alunos e fazer com que aprendam, ainda mais àqueles que nasceram na era digital. Segundo [Prensky 2001], os alunos nativos digitais aprendem de uma forma diferente, essa geração tem acesso a informação de forma rápida e aprende fazendo. Segundo o autor, é preciso que as escolas promovam novas mecânicas de ensino que sejam diferentes das tradicionais, nas quais o aluno é um ouvinte que recebe as instruções dos professores.

Neste cenário, a gamificação aparece como uma ferramenta capaz de auxiliar tanto o professor quanto o aluno, pois ajuda na implementação de uma forma de ensino mais dinâmica, intuitiva e fluída [Massário et al. 2019].

2.2. Jogos educacionais

Os jogos educacionais digitais são uma das novas formas de ensino proporcionadas pela massificação do acesso a Informática. Eles permitem a gamificação do processo de ensino-aprendizagem, fazendo com que o conhecimento seja construído através de atividades que incentivam o jogador a pensar, podendo até em alguns exemplos se moldar ao seu perfil, criando assim uma experiência de aprendizagem única que atenda às necessidades de cada aluno.

Como a aprendizagem se dá por meio de um jogo, o jogador (aluno) assume um papel atuante, no qual ele deve pensar no que fazer, não se limitando apenas a reproduzir o que estudou de forma teórica passando assim a pensar. “Os jogos forçam o aluno a decidir, a escolher, a priorizar. Todos os benefícios intelectuais do jogo resultam dessa virtude fundamental, porque aprender como pensar, em última análise, tem a ver com aprender a tomar a decisão certa: pesar a evidência, analisar situações, consultar suas metas em longo prazo, e então, decidir” [Johnson 2005].

Outro aspecto importante dos jogos digitais é o apelo motivacional dado o seu caráter lúdico, imersivo e experiencial que proporcionam situações de aprendizagem mais divertidas, desafiadoras, despertando a descoberta, a fantasia e a curiosidade [Gomes e De Melo 2016]. O jogo funciona como um estímulo para que a experiência de ensino-aprendizagem continue, mesmo após o jogador fazer uma pausa, extrapolando os limites da sala de aula.

2.3. Expressões Lógicas

De acordo com [Berg 2007], o conceito de expressão, em termos computacionais, está intimamente ligado ao conceito de expressão (ou fórmula) matemática, em que um conjunto de variáveis e constantes numéricas relaciona-se por meio de operadores, compondo uma fórmula que, uma vez avaliada, resulta num valor.

Com estes operadores é possível, entre outras coisas, criar expressões de armazenamento de valores em variáveis e Expressões Lógicas que guiam o fluxo de execução de uma determinada aplicação. Desta forma, se torna essencial que alunos de cursos de Computação dominem a utilização desses assuntos, entendendo o seu funcionamento, para que consigam relacionar soluções de Programação que exigem o conhecimento de portas lógicas e tabelas-verdade, assuntos importantes de disciplinas como Engenharia de Hardware e Lógica de Programação.

2.4. Trabalhos Relacionados

Existem diversas pesquisas relacionadas ao ensino de Programação, visto que se trata de uma disciplina introdutória nos cursos de Computação e que, comumente, não está presente nas grades curriculares em séries que antecedem a entrada do aluno no ensino superior ou em cursos profissionalizantes. Neste sentido, serão apresentadas algumas pesquisas relacionadas.

A busca pelos trabalhos relacionados foi feita inicialmente através de pesquisa utilizando o portal de periódico da Capes, com as seguintes palavras chaves: “jogos educacionais” e “ensino programação”. Dos trabalhos encontrados foi utilizada a técnica de revisão sistemática de literatura *snowballing*, para encontrar novos trabalhos. Dos resultados encontrados foram excluídos todos os trabalhos que tivessem sido publicados há mais

de 10 anos, deste filtro foram selecionados aqueles que tivessem realizado algum tipo de avaliação do *software* proposto. Posteriormente foi feita a exclusão de trabalhos que possuíam ferramentas (jogos) que não foram desenvolvidas pelos autores. Nesta seção apresentam-se aqueles em que os pesquisadores consideraram mais relevantes.

AgentJ (Figura 1) é um *framework* para o ambiente *desktop* que busca diminuir a dificuldade de ensino e aprendizagem de lógica de programação. [Vahldick e Pereira 2019] apresentam a ferramenta como uma melhoria de um trabalho anterior dos autores, o Furbot [Vahldick e Mattos 2008], que consistia em um ambiente onde os alunos deveriam programar um robô para explorar e coletar itens em um mundo virtual.

Nesta nova versão foi feita a implementação de um cenário visual, além de uma página com exercícios a serem realizados e a inserção de objetivos a serem cumpridos para terminar a resolução do problema proposto.

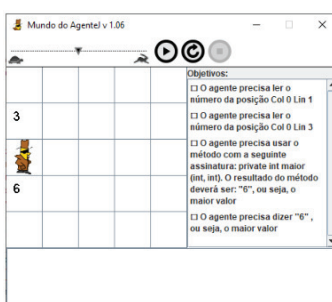


Figura 1. Mundo Visual do AgenteJ apresentando os objetivos do exercício. Fonte: [Vahldick e Pereira 2019]

O *framework* foi desenvolvido utilizando a linguagem de programação Java, com os dados referentes as atividades sendo guardados em arquivos XML. Para que fosse possível fazer a verificação dos códigos criados pelos alunos, os desenvolvedores também fizeram uso da API de Reflexão Java, além da biblioteca Javaassist. Para avaliar a aplicação, os autores realizaram um experimento com uma turma da disciplina de Introdução a Programação, a qual foi dividida em dois grupos: o primeiro deveria utilizar o *framework* com o novo recurso (o incremento de objetivos parciais na resolução dos problemas propostos), por sua vez, o restante dos alunos usaria a versão antiga do software. Após realizar o teste da aplicação, o primeiro grupo respondeu um questionário para avaliar sua experiência e dar suas opiniões. Como resultados, os autores puderam constatar que quem fez uso da versão com as melhorias conseguiu estatisticamente resolver mais exercícios e através do questionário foi possível identificar uma grande aceitação dos estudantes à nova versão da aplicação.

KLouro (Figura 2) [Silva e Dantas 2014] é um Objeto de Aprendizagem (OA), do tipo jogo, que objetiva ajudar alunos a aprender de uma melhor forma os operadores lógicos e aritméticos da linguagem de programação Python [Foundation 2020], além de analisar Expressões Lógicas. Os autores também realizaram a avaliação do protótipo desenvolvido. Durante o trabalho, foi desenvolvido um protótipo do jogo para o ambiente *web*, utilizando a ferramenta *Construct 2*, a qual permite o desenvolvimento para diversas plataformas sem a necessidade de codificar efetivamente. O desenvolvimento conside-

rou princípios técnicos, pedagógicos e de *game design* encontrados após pesquisa bibliográfica. Depois da conclusão do protótipo, foi realizada uma avaliação do jogo seguindo abordagens comportamentalistas e construtivistas propostas por [Reategui e Finco 2010], na qual os desenvolvedores verificaram se o KLouro seguia os critérios técnicos definidos pelos autores. Também foram observados princípios de *game design*. Por fim, os autores realizaram a aplicação de um formulário *online* com alunos da disciplina introdutória de Programação, que avaliaram o jogo, no entanto, é importante destacar a participação de apenas três indivíduos, sendo assim uma amostra muito pequena para se tirar qualquer conclusão. Os pesquisadores concluíram que projetar o *software* a partir de aspectos pedagógicos, técnicos e de *game design* se mostrou positivo e acreditam que ao chegar ao fim do desenvolvimento, o objetivo será atingido. Os autores apontam o fato de não conseguirem concluir o desenvolvimento do jogo, propondo sua finalização em trabalhos futuros, assim como a adaptação do jogo para o ambiente *mobile*, além de avaliar a percepção da utilidade do *game* pelos alunos.



Figura 2. KLouro - Missão da Fase 1. Fonte: [Silva e Dantas 2014]

Sortia 2.0 (Figura 3) [Battistella et al.] é um jogo de ordenação de dados desenvolvido para que os alunos da disciplina de Estrutura de Dados pudessem ter contato de forma prática com o processo de ordenação de números inteiros. Na pesquisa, os autores realizaram a apresentação e a avaliação do jogo. Para o desenvolvimento do jogo foi feito inicialmente um protótipo de baixa fidelidade (com caneta e papel) das telas, seguindo regras de *design* de jogos. Após a aprovação, foi feito um segundo protótipo, dessa vez com alta fidelidade. Por fim, realizaram a codificação utilizando JavaScript, PHP e HTML5. O jogo foi finalizado e disponibilizado gratuitamente na *internet*. Após o desenvolvimento foi realizado um estudo de caso seguindo o modelo de avaliação de jogos educativos MEEGA [Savi et al. 2011], no qual avaliam-se três parâmetros: motivação, experiência do usuário e aprendizagem. Com base nos resultados da avaliação da ferramenta, foi possível constatar que os alunos consideraram o jogo relevante, satisfatório e que o mesmo contribuiu para o processo de aprendizagem, além de se mostrar atrativo e desafiador. Como trabalhos futuros, os autores propõem a aplicação do jogo em outras disciplinas de cursos de Computação e também será desenvolvida uma versão do jogo *online* com o algoritmo *Quicksort*.



Figura 3. Sortia 2.0. Fonte: [Battistella et al.]

2.4.1. Análise Comparativa

A Tabela 1 apresenta uma análise comparativa dos trabalhos relacionados e o *Boolean Defuse*, objeto principal deste artigo. Foram analisadas cinco características: tecnologia, plataformas, assuntos abordados, avaliação e público alvo.

Tabela 1. Análise comparativa dos trabalhos relacionados

| Estudo | Tecnologia | Plataformas | Assuntos abordados | Avaliação | Público alvo |
|----------------|------------|-------------|------------------------|--|----------------------|
| Agentj | Java | Desktop | Lógica de Programação | Avaliação Qualitativa | Alunos de Graduação |
| | XML | | | Teste para Comparação de Duas Médias Populacionais | |
| | | | | Questionário | |
| | | | | Experimentação | |
| Klouro | Construct2 | Web | Expressões Lógicas | Avaliação de AO Proposta por Reategui et. al. (2010) | Alunos de Graduação |
| | | | Expressões Aritméticas | Questionário | |
| | | | Estruturas de controle | | |
| Sortia 2.0 | HTML5 | Web | Algoritmo de Ordenação | Modelo de Avaliação de Jogos educacionais MEEGA | Alunos de Graduação |
| | JavaScript | | | | |
| | PHP | | | | |
| Boolean Defuse | Java | Desktop | Expressões Lógicas | Modelo de Avaliação de Jogos educacionais MEEGA | Alunos de Computação |
| | XML | | | SUS | |
| | | | | Inferência Estatística | |

O *Boolean defuse* se assemelha ao AgentJ quanto a tecnologia utilizada e a plataforma, ambos são desenvolvidos para *desktop* utilizando a linguagem de programação Java e arquivos XML como base de dados. Enquanto os jogos KLouro e Sortia 2.0 foram desenvolvidos para ambiente *web* e não possuem nenhuma base de dados aberta para a inserção de novos desafios. Entretanto, AgentJ e *Boolean Defuse* permitem de forma mais simples a inserção de novas questões após a conclusão de seu desenvolvimento.

Ao analisar os assuntos abordados, percebe-se que apesar de todos os jogos terem como público-alvo alunos de graduação, cada um foca em um tema diferente e para pessoas com níveis de conhecimento de Programação diferentes. O Sortia 2.0 pressupõe que o jogador já tem certo conhecimento de Programação, enquanto os demais são focados em alunos iniciantes. Ainda entre os jogos que abordam Lógica de Programação, existe a diferença da abrangência de assuntos: *Boolean Defuse* e KLouro possuem assuntos específicos; e AgentJ cobre uma gama maior de temas.

Por fim, quanto ao método de avaliação dos trabalhos, todos apresentam sessões de teste com usuários. No entanto, é importante destacar que o teste realizado em KLouro, possuía um número muito pequeno de participantes, apenas três. Sortia 2.0 utiliza o método MEEGA e *Boolean Defuse* utiliza parte de MEEGA, bem como da utilização da técnica SUS para avaliação de design e testes de inferência estatística para verificar hipóteses.

3. Método

O desenvolvimento do jogo teve como base o *feedback* da avaliação realizada na primeira versão do *Boolean Defuse* [Nogueira et al. 2020] e seguiu o processo de oito fases definido por [Novak 2012]: Conceito, Pré-produção, Protótipo, Produção, *Alpha*, *Beta*, Ouro e Pós-Produção.

Inicialmente, foi definido o conceito do jogo *Boolean Defuse*, através de uma definição formal, que consistiu na criação de um documento que apresentava o objetivo, o assunto que o jogo abordaria, público alvo, levantamento dos recursos disponíveis (pessoal, equipamentos, etc) para o desenvolvimento, além de uma descrição das principais mecânicas por meio de *storyboard*.

Na segunda fase, foi dado início a fase de Pré-produção, em que foram definidos qual o estilo de arte utilizado no jogo, além do planejamento de como aconteceria o desenvolvimento e quais as tecnologias seriam utilizadas.

O jogo foi desenvolvido com gráficos 2D e para o ambiente *desktop*, com isso para a codificação do jogo foi utilizada a linguagem de programação Java [Oracle 2009]. O padrão de arquitetura de software utilizado foi o *Model-View-Controller* (MVC) [Krasner e Pope 1988], por facilitar a manutenção, reusabilidade do código para o caso de desenvolvimento de versões do jogo para outras plataformas como a Web. Para o armazenamento de dados relacionados ao jogo foram utilizados arquivos XML, onde seriam salvas informações relativas à elementos do cenário do jogo, para acessar as bases de dados foi utilizado o padrão DAO (*Data Access Object*).

Para verificar se a mecânica e o estilo de arte definidos nas fases anteriores eram atrativos e divertidos, foram realizados *mockups* do jogo. Depois da validação, foi dado início a fase de Produção. Nessa fase, inicialmente, realizou-se a produção artística, que consistiu na criação de *assets* utilizando uma ferramenta de desenho vetorial e na busca de sons para o jogo. Em seguida foi dado início a codificação.

Quando o desenvolvimento do jogo contemplou a finalização da interface gráfica e sua *engine*, foi iniciada a fase *Alpha*, em que o jogo foi testado pelo desenvolvedor, com a finalidade de identificar *bugs* e realizar as devidas correções. Com o jogo em uma versão estável e menor número de erros, foi realizada a fase *Beta*, que consistiu em um teste com usuários.

Atualmente vive-se a pandemia do COVID-19 [WHO 2021] que tem como consequência o isolamento social, não permitindo o encontro de indivíduos em um mesmo local, como, por exemplo, laboratórios de Computação, impossibilitando a realização da avaliação do jogo em um experimento controlado. Dito isso, a pesquisa de avaliação de *Boolean Defuse* se deu de forma *online*, pelo método prospectivo por questionário através

da utilização de formulário¹ criado com auxílio da ferramenta *Google Forms* e distribuído por convite enviado a e-mails de grupos e mídias sociais, assumindo-se uma amostragem não-probabilística e por conveniência.

O formulário é composto por 56 questões entre abertas e fechadas, formuladas de maneira positivo-afirmativa, exceto as perguntas para caracterização de perfil. As respostas para as objetivas eram dispostas em escala Likert [Likert 1932] de 5 pontos, variando de discordo totalmente até concordo totalmente.

O questionário ficou organizado da seguinte forma: (i) 2 questões sobre aceitação voluntária na pesquisa e contato; (ii) 5 questões sobre caracterização do indivíduo; (iii) 9 perguntas para análise de percepção por autoavaliação de aprendizagem de Expressão Lógica antes de jogar o jogo. Essas mesmas perguntas são utilizadas para analisar o depois de jogar o jogo; (iv) orientação de tarefas a realizar; (v) 10 questões sobre usabilidade de design; (vi) 10 questões sobre usabilidade pedagógica; (vii) 9 questões a partir da abordagem MEEGA das quais 4 referentes ao efeito motivador e 5 questões sobre experiência do usuário; e (viii) 2 questões para atribuição de uma nota e sugestões e/ou comentários para o jogo a partir da experiência vivenciada.

São perguntas pertencentes ao questionário: **Caracterização de perfil:** Q1-Qual sua faixa etária?, Q2-Qual seu Sexo?, Q3-Você tem alguma familiaridade com Expressões Lógicas?, Q4-Você costuma jogar jogos digitais?, Q5-Qual o seu grau de instrução?; **Análise da percepção de conhecimento:** Q6-Lembrar o que são Operadores Lógicos, Q7-Compreender como funcionam Operadores Lógicos, Q8-Aplicar na prática Operadores Lógicos, Q9-Lembrar qual a ordem de execução de operações lógicas, Q10-Compreender como funciona a ordem de execução de operações lógicas, Q11-Aplicar na prática a ordem de execução de operações lógicas, Q12-Lembrar o que é Lógica Booleana, Q13-Compreender como funciona Lógica Booleana, Q14-Aplicar na prática Lógica Booleana; **Usabilidade design):** Q15-O jogo é de fácil manuseio, Q16-O jogo é compreensível, Q17-Foi fácil aprender a usar o jogo, Q18-O jogo possui ícones que ajudam a impedir a ocorrência de erros, Q19-O jogo possui instruções, ações e opções visíveis ou facilmente recuperáveis sempre que apropriado para o uso, Q20-O usuário transita entre as telas do jogo com facilidade, Q21-O usuário é mantido informado sobre o que está acontecendo, Q22-É fornecido feedback adequado e em tempo razoável sobre a ação do usuário, Q23-O jogo exibe o necessário para o usuário, Q24-O jogo NÃO apresenta informações em excesso; **Usabilidade pedagógica:** Q25-O jogo é eficiente para a sua aprendizagem, comparando-o com outras atividades da disciplina, Q26-O jogo contribui para a aprendizagem do conteúdo, Q27-Ficou claro como o conteúdo do jogo está relacionado com coisas já sabidas, Q28-O conteúdo é coerente e contextualizado com a área e o nível de ensino propostos, Q29-Aprender o jogo não é chato, mas sim divertido, Q30-Jogadores permanecem atentos a tarefas importantes, Q31-Eu jogaria o jogo novamente, Q32-É possível usar o conhecimento adquirido no jogo em uma situação prática, Q33-O jogo dá suporte às interações entre o jogador e a máquina, Q34-O jogo oferece *feedback* construtivo, permitindo identificar claramente acertos e erros; **Reação em relação ao efeito motivador do jogo:** Q35-O funcionamento deste jogo está adequado ao meu jeito de aprender, Q36-Foi fácil entender o jogo e começar a utilizá-lo como material de estudo, Q37-Ao passar pelas etapas do jogo senti confiança de que estava aprendendo, Q38-

¹<https://forms.gle/rZMhmCR1gSfQm25A8>

É por causa do meu esforço pessoal que consigo avançar no jogo; **Reação em relação a experiência do usuário:** Q39-Eu jogaria o jogo novamente, Q40-Este jogo é adequadamente desafiador para mim, as tarefas não são muito fáceis nem muito difíceis, Q41-Eu recomendaria este jogo para meus colegas, Q42-Consegui atingir os objetivos do jogo por meio das minhas habilidades, Q43-O jogo evolui num ritmo adequado e não fica monótono.

As perguntas de concordância, caracterização de perfil e orientações de tarefas, deveriam ser respondidas antes de jogar Boolean Defuse, bem como da primeira autoavaliação sobre percepção de aprendizagem antes de jogar o jogo. Os demais itens do formulário deveriam ser respondidos apenas após o uso do jogo. Para isso, o participante deveria realizar as tarefas, a saber: (i) Baixe o jogo *Boolean Defuse* - Versão .EXE: encurtador.com.br/fBTW5; (ii) Descompacte o arquivo baixado; (iii) Execute o jogo (duplo clique no arquivo BooleanDefuse.exe ou BooleanDefuse.jar a depender da versão baixada); e (iv) Jogue Boolean Defuse.

A *feedback* de realização das tarefas é dada por caixa de seleção e caso o usuário não tenha as marcadas suas respostas seriam desconsideradas, em especial, a última, uma vez que as demais tarefas poderiam ser resolvidas de outras formas alheias a explicação do formulário.

As perguntas presentes no questionário foram baseadas nos trabalhos de [Savi et al. 2011, Oliveira e Savoine 2011, Ferreira et al. 2014, Neto et al. 2015, Júnior et al. 2017, Vasconcelos 2017, Silva et al. 2020], fazendo adequações para atender a avaliação pretendida. Os questionamentos sobre efeito motivador, experiência ao jogar e a percepção de ganho de aprendizagem, em especial, foram extraídas do modelo de avaliação de jogos educacionais MEEGA [Savi et al. 2011].

Antes de realizar a avaliação de *Boolean Defuse*, primeiramente, verifica-se a confiabilidade dos dados (respostas dos participantes) através do coeficiente (α) Alfa de Cronbach [Cronbach 1951]. Para [George e Mallery 2003], o valor de α indica a qualidade dos dados, podendo variar entre 0 e 1, com a seguinte classificação para os valores de α : $\alpha > 0.9$ – Excelente; $\alpha > 0.8$ – Bom; $\alpha > 0.7$ – Aceitável; $\alpha > 0.6$ – Questionável; $\alpha > 0.5$ – Ruim; e $\alpha < 0.5$ – Inaceitável. Em seguida, tem-se: (i) análise descritiva dos dados; (ii) para avaliação de qualidade de software: *System Usability Scale* (SUS) [Bangor et al. 2009]; e (iii) inferência estatística a partir de testes de hipóteses por Qui-Quadrado, ANOVA, e Teste-t de *Student*.

Segundo [Bangor et al. 2009], SUS permite averiguar o nível de usabilidade de um sistema, com pontuação variando entre 0 e 100 associada a uma escala de conceitos e adjetivos, os quais são ilustrados na Figura 4.

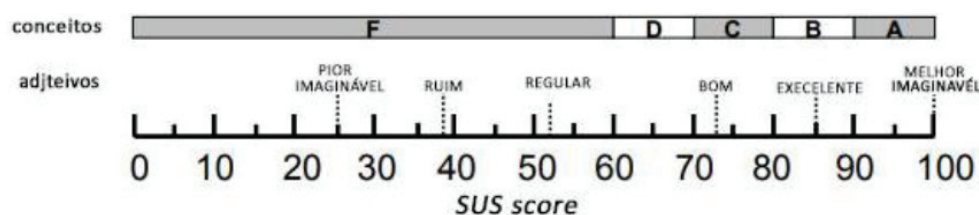


Figura 4. Classificação SUS. Fonte: [Bangor et al. 2009]

Para [Revythi e Tselios 2017], é um dos métodos mais confiáveis e válidos para se medir a usabilidade de uma determinada interface pelo usuário, mesmo que sejam utilizadas pequenas amostras. Segundo [Brooke 1996], seu questionário é baseado em 10 perguntas relacionadas a princípios de *design*, alternadas entre afirmativas e negativas, que resultam em um método cientificamente apurado em um espaço de tempo relativamente curto e que para obtenção da pontuação SUS, tem-se: (i) para as questões ímpares (1, 3, 5, 7 e 9), subtrai-se 1 da pontuação atribuída; (ii) em questões pares (2, 4, 6, 8 e 10) subtrai-se a resposta de 5; (iii) somam-se todos os valores das dez perguntas e multiplica-se por 2.5, tendo como resultado a pontuação final de um usuário; e (iv) a pontuação final do software é tida pela média das pontuações de todos os usuários. A Figura 5 ilustra o fluxograma para obtenção da pontuação SUS.

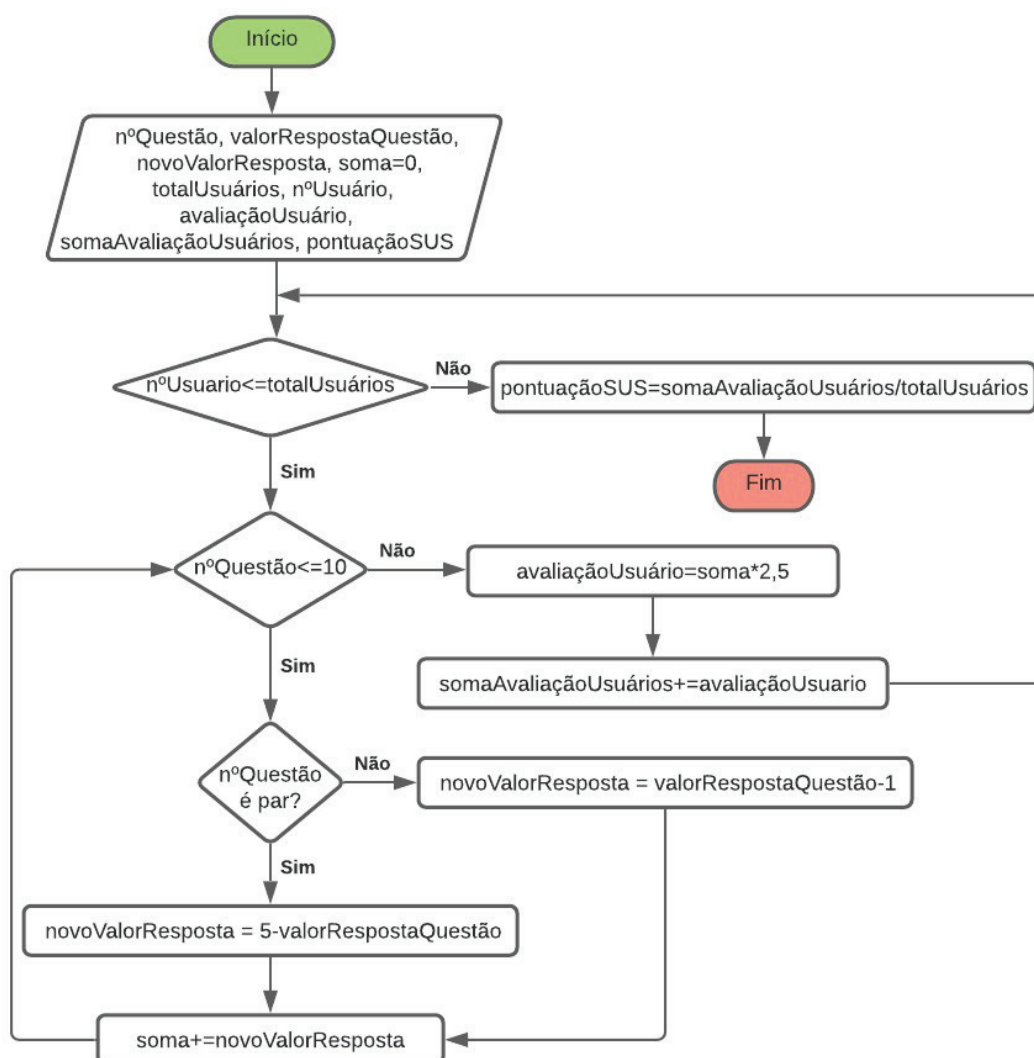


Figura 5. Fluxograma do método SUS

Em SUS alternam-se as perguntas entre afirmativas e negativas. No entanto, as perguntas desta pesquisa estão formuladas de maneira positivo-afirmativa, por os pesqui-

sadores entenderem que dessa forma a assimilação entre concordância e discordância é mais natural. Sendo assim, antes de continuar a avaliação, as respostas pares do questionário de usabilidade de *design* foram ajustadas (Figura 6), ou seja, uma resposta com valor 5 (Concordo Totalmente) é convertida em 1 (Discordo Totalmente); 4 em 2; 3 continua 3; 2 em 4; e 1 em 5, consequentemente, refletindo a opinião inversa caso o questionamento estivesse escrito de maneira positivo-negativa.

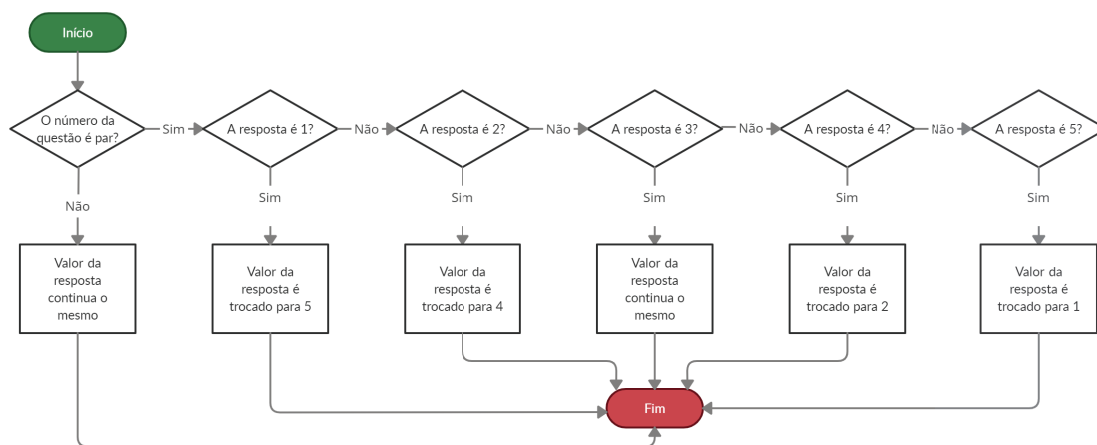


Figura 6. Fluxograma do método para inversão de resposta de pergunta par

Para a análise por inferência estatística na avaliação de *Boolean Defuse* tem-se a verificações de pressupostos por testes de hipóteses, definidos a partir de quatro situações. Na Situação 1 (S1), analisa se há diferença significativa entre as médias das avaliações de Experiência Geral em relação a SUS, a partir das hipóteses: (i) hipótese nula (H_{0-S1}) – as médias das notas para Experiência Geral e SUS são consideradas estatisticamente iguais; e (ii) hipótese alternativa (H_{1-S1}) – as médias das notas para Experiência Geral e SUS diferem estatisticamente.

Na Situação 2 (S2) é verificada a contribuição de *Boolean Desufe*. Para isso será analisada se há relação de dependência entre os níveis de aceitação em relação à avaliação do jogo e se confirmada essa dependência, então se verificam as seguintes hipóteses definidas: (i) hipótese nula (H_{0-S2}) – o jogo educacional *Boolean Defuse* não contribui para o processo de aprendizagem de Expressões Lógicas; e (ii) hipótese alternativa (H_{1-S2}) – o jogo educacional *Boolean Defuse* contribui para o processo de aprendizagem de Expressões Lógicas.

Visando analisar se há relação de dependência de variáveis de caracterização de indivíduo em relação à avaliação de *Boolean Defuse*, verifica-se na Situação 3 (S3) as seguintes hipóteses: (i) hipótese nula (H_{0-S3}) – avaliação do jogo educacional *Boolean Defuse* é independente de variável de caracterização é indivíduo; e (ii) hipótese alternativa (H_{1-S3}) – avaliação do jogo educacional *Boolean Defuse* é dependente de variável de caracterização é indivíduo.

Por fim, na Situação 4 (S4) é verificada a existência de diferença estatística significativa entre a percepção de aprendizagem do usuário antes e depois da utilização do jogo. Para isso, são verificadas as hipóteses: (i) hipótese nula (H_{0-S4}) – as médias das percepções de aprendizagem antes e depois da utilização do jogo educacional *Boolean*

Defuse são consideradas estatisticamente iguais; e (ii) hipótese alternativa (H_{1-S4}) – as médias das percepções de aprendizagem antes e depois da utilização do jogo educacional *Boolean Defuse* diferem estatisticamente.

Para responder a Situações 1, tem-se a Análise de Variância (ANOVA) a partir do Teste F para verificação das hipóteses [Vieira 2006]. Para as Situações 2 e 3 será utilizado o teste hipóteses por Qui-Quadrado χ^2 [Plackett 1983 apud [Levin e Fox 2004]].

Segundo [Siegel 1975], o teste de Qui-Quadrado só pode ser aplicado, em tabelas rxk , (várias linhas x várias colunas), caso o número de frequências esperadas com valor abaixo de 5 seja inferior a 20% do total das frequências esperadas e se nenhum dos valores seja inferior a 1. Caso contrário o pesquisador deverá, para realizar o teste, fazer um reagrupamento dos dados, ou seja, as tabelas de contingências com categorias combinadas a partir das variáveis de modo a aumentar as frequências esperadas.

Por fim, para verificar a Situação 4, se a média das percepções dos usuários quanto ao nível de aprendizagem sobre Expressões Lógicas antes e depois da utilização do jogo *Boolean Defuse* sofreram alguma alteração estatisticamente significativa, tem-se o teste T de *Student* [Student 1908]

Para concluir a fase *Beta*, trabalhos futuros, serão realizadas as correções dos problemas encontrados no jogo com base nos resultados dos testes realizados pelos usuários. Ao fim dessa fase, o jogo já estará finalizado e pronto para ser disponibilizado para todos os usuários sendo assim entrará na fase Ouro. Com o jogo concluído as atenções do desenvolvedor deverão se voltar para a pós-produção, período no qual serão criados conteúdos e atualizações que permitam que o jogo tenha um período de vida longo.

O detalhamento de todos os métodos e resultados obtidos da pesquisa é encontrado em [Nogueira e D'Emery 2021]

4. Resultados

4.1. Boolean Defuse 2.0

A ferramenta educacional foi idealizada com o objetivo de auxiliar alunos de cursos da área de Computação na prática de Expressões Lógicas. O jogo consiste em desarmar uma bomba através da resolução de *puzzles*, mecânica semelhante à *Them Bombs e Keep talking and nobody explodes*. Porém, a diferença é que no jogo proposto os desafios a serem solucionados devem testar os conhecimentos dos aluno sobre o tema.

Como o *Boolean Defuse* se trata de um jogo educacional, a escolha dos desafios presentes na bomba, os quais são chamados de módulos, passou inicialmente por um levantamento das diferentes formas como o conhecimento do tema é comumente abordado, sendo selecionadas: resolução de tabelas verdade, problemas com expressões lógicas e blocos de programação. Após definidas as habilidades que deveriam ser testadas, realizou-se um levantamento de desafios presentes em *games* correlatos que são focados apenas no lúdico. Deste modo, foram selecionados módulos que seguem as seguintes mecânicas: cortar fios, substituição de palavras por símbolos, busca de palavras chaves e traduções.

A partir dessas definições, foram criados módulos em que cada ação do jogador implica no desarme do módulo ou na detonação do explosivo. São módulos:(i) **Fios**: este

módulo é composto por três fios os quais o jogador deverá decidir se deve ou não cortá-los. A decisão é baseada através da resolução de expressões lógicas associadas a cada um dos fios, caso a expressão seja verdadeiro o fio deverá ser cortado e caso contrário deverá permanecer intacto; (ii) **Quiz:** também envolve a resolução de uma expressão lógica, no entanto, a mecânica empregada é a de substituir palavras dispostas em um *display* por símbolos lógicos. A solução da expressão deve ser associada ao clique dos botões verdadeiro ou falso; (iii) **Teste de Mesa:** esse módulo consiste em realizar o teste de mesa de um código-fonte específico e inserir a saída deste em campos especificados; e (iv) **Morse:** neste módulo deve-se traduzir alguns textos escritos em código morse, cada tradução deve ser inserida em campo específico, ao final tem-se uma expressão lógica cujo resultado deve também ser inserido em campo especificado.

A versão 2.0 do jogo educacional *Boolean Defuse* (Figuras 7 e 8) é aprimoramento de melhorias decorrentes da avaliação da primeira versão [Nogueira et al. 2020]. Tem-se como principal diferença, a possibilidade do jogo ser realizado em modo *singleplayer* a partir da interação com um personagem não-jogável (NPC, do inglês *non-player character*), comumente utilizado em jogos do tipo *Role-Playing Game*. Esta mudança se deve ao fato de na primeira versão, ser demandada a colaboração de dois jogadores, Operador e Especialista, um com acesso apenas à bomba mas sem saber o procedimento de desarmá-la e outro as instruções de desarmamento, mas sem acesso à bomba, respectivamente. Com isso demanda um processo de comunicação entre os jogadores. Agora, na versão 2.0, o jogador deve tentar desarmar a partir das instruções e interações com o desafiante NPC Jigsaw.



Figura 7. Abertura de *Boolean Defuse*

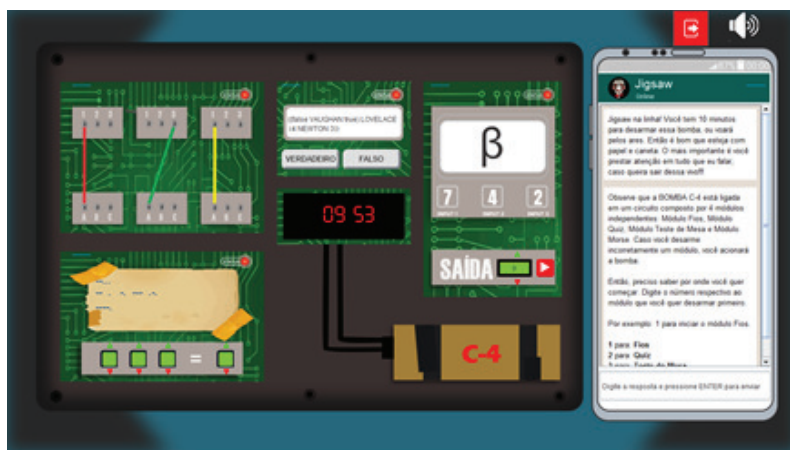


Figura 8. Bomba de *Boolean Defuse*

4.2. Análise dos Dados

A avaliação de *Boolean Defuse* utiliza as respostas de 24 indivíduos participantes da avaliação do jogo. A amostra é composta por 20 (83,33) homens e 4 (16,67) mulheres; com maioria entre 20 e 29 anos (63%), 12% entre 15 a 19 anos, 17% entre 30 a 39 anos e 8% entre 40 a 49 anos; todos os indivíduos possuem familiaridade com Expressões Lógicas; 83% costumam jogar jogos digitais; e para o grau de instrução tem-se: 54% com nível superior incompleto; 21% superior completo; 17% com pós-graduação, 4% ensino fundamental completo e 4% ensino médio completo. A existência de indivíduos da educação básica está atrelada ao fato destes terem em suas grades curriculares disciplinas relacionadas a Programação de Computadores.

A partir das respostas desses indivíduos, realiza-se a avaliação de *Boolean Defuse*, mas antes verifica-se a confiabilidade quanto à qualidade dos dados.

A Tabela 2 apresenta o resultado do coeficiente de Alfa de Cronbach [Cronbach 1951] para cada um dos critérios a serem analisados na avaliação.

Tabela 2. Alfa de Cronbach para as respostas dos usuários

| Critério | Nº de Questões (n) | Nº de Participantes | Total de respostas | \alpha | Qualidade |
|------------------------------------|--------------------|---------------------|--------------------|--------|-----------|
| Usabilidade de <i>design</i> | 10 | 24 | 24 | 0,878 | Bom |
| Usabilidade pedagógica | 10 | 24 | 240 | 0,903 | Excelente |
| Usabilidade | 20 | 24 | 480 | 0,940 | Excelente |
| Motivação e experiência de usuário | 9 | 24 | 216 | 0,912 | Excelente |
| Percepção Aprendizagem | 18 | 24 | 432 | 0,917 | Excelente |

Portanto, afirma-se que os dados levantados por meio do questionário são tidos como um instrumento de medida confiável por possuírem confiabilidade alta.

Após constatar a confiabilidade dos dados, verifica-se a aceitação dos critérios de avaliação adotados. De maneira geral, a análise dos gráficos (Figuras 9, 10, 11 e 12) aponta aceitação, em pontos percentuais, em todos os aspectos avaliados pelos usuários. Para as análises gráficas, as respostas estão agrupadas em três grupos de aceitação: concordância (4 e 5), neutro (3) e discordância (1 e 2).

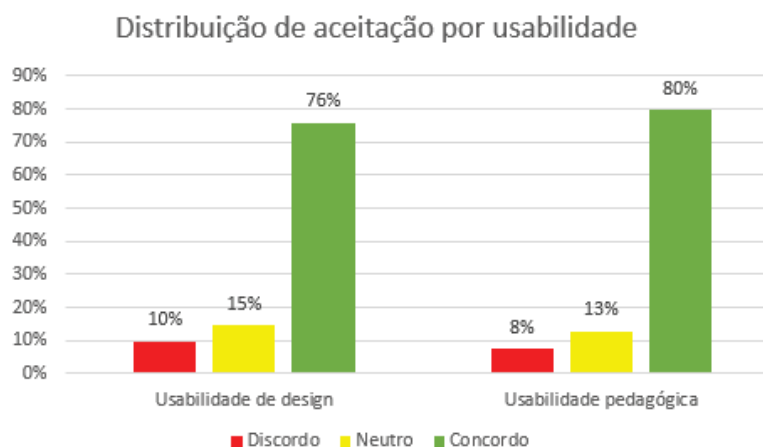


Figura 9. Aceitação de critérios de usabilidade de design e pedagógica

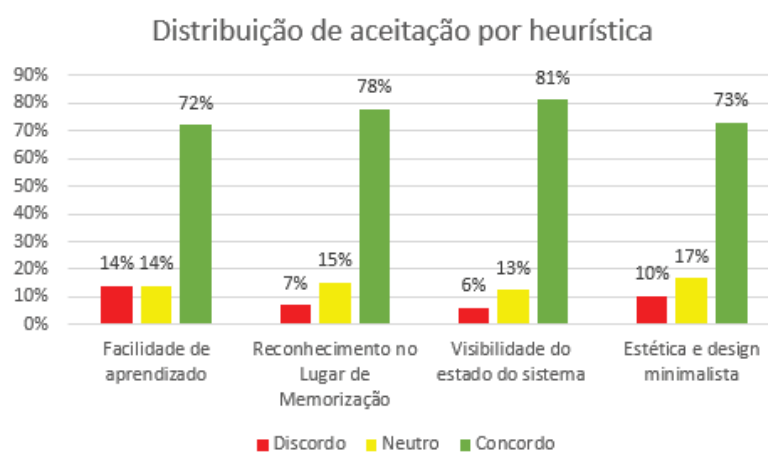


Figura 10. Aceitação das heurísticas empregadas na usabilidade de design

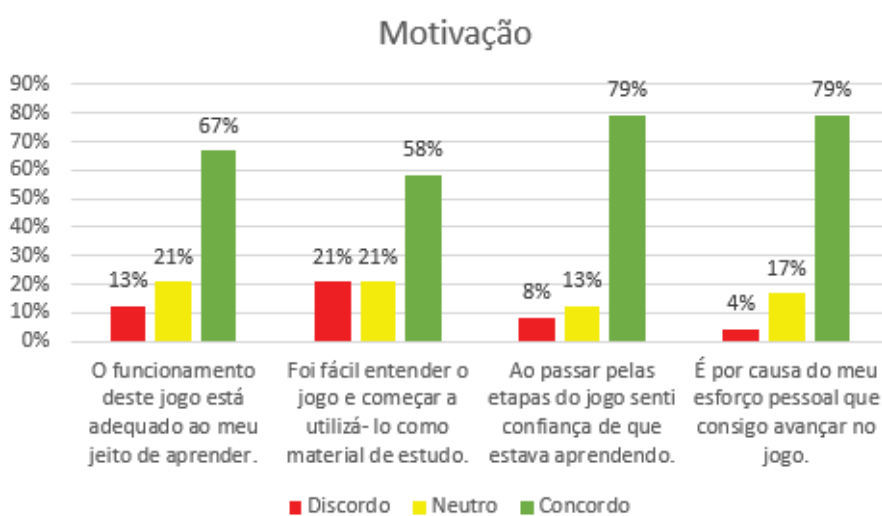


Figura 11. Aceitação dos aspectos de motivação

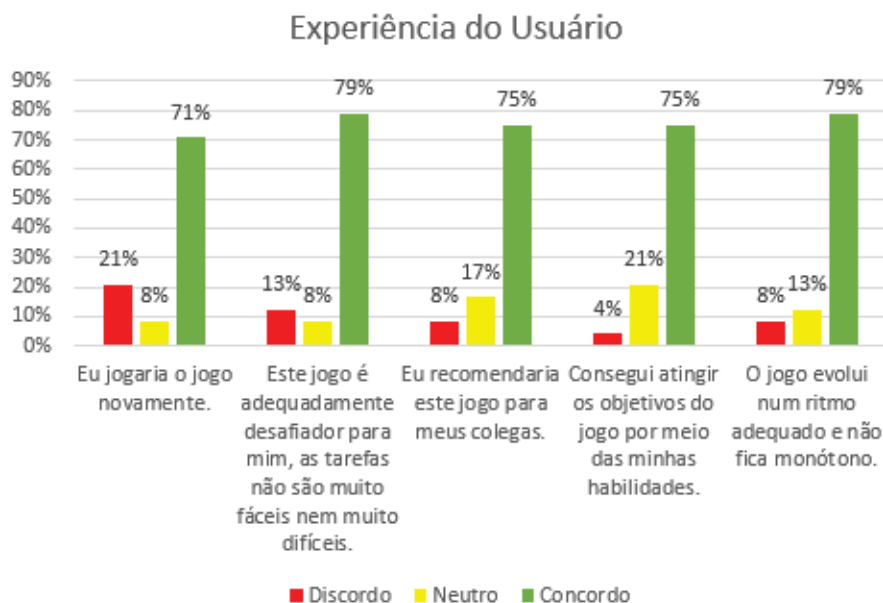


Figura 12. Aceitação dos aspectos de experiência do usuário

As usabilidades de design e pedagógicas (Figura 9) obtiveram aceitações de 76% e 80%, respectivamente; e ainda que analisadas separadamente as heurísticas de [Nielsen 1994] adotadas (Figura 10), – facilidade de aprendizado, reconhecimento no lugar de memorizar, visibilidade do estado atual do sistema e estética e design minimalista – observam-se margens de aceitação superiores a 70% às heurísticas aplicadas no projeto de *design*.

Apesar dos aspectos motivacionais obterem maioria de aceitação, pouco mais da metade, 58%, apontaram ser fácil entender o jogo e começar a utilizá-lo como material de estudo; experiência do usuário também aponta bons níveis de aceitação, com todos questionamentos apresentando porcentagens superiores a 70%. Como recomendado em MEEGA [Savi et al. 2011], a distribuição das concordâncias das questões referentes a motivação (Figura 11) e experiência do usuário (Figura 12) foram colocadas separadamente, de maneira a melhorar a análise dos pontos específicos e como estes possam ser melhorados. Sendo assim, destaca-se que neutralidade (21%) é considerada aspecto positivo, elevando o índice de aceitação, e que 21% dos participantes da avaliação apontaram ser difícil entender o jogo e começar a utilizá-lo como material de estudo. A

Ao analisar esse comportamento a partir do aspecto qualitativo dos comentários disponibilizados no formulário, alguns comentários e sugestões apontam a possibilidade de flexibilização quanto às dificuldades, como, por exemplo, mudança de nível de dificuldade, diminuir o circuito da bomba, cada módulo ser uma bomba, atribuir níveis de desarme e caso um módulo seja desarmado o usuário poderia continuar desse ponto. Ainda que as sugestões sejam interessantes do ponto de vista acadêmico, os autores destacam o fato de o cenário ser uma bomba, ou seja, uma ação errada implica em explodi-la. Fato este observado, quando analisado os 13% que apontam discordância ao seu jeito de aprender, a análise qualitativa aponta o fato da pressão imposta pelo cenário da bomba, uma vez que o usuário teria um tempo determinado para desarmá-la, e neste sentido su-

gestões para mudança de nível de dificuldade com a flexibilização de tempo de desarme fizessem mais sentido.

4.3. Avaliação de Qualidade e Análise e Interpretação por Inferência Estatística

Para as análises, a Tabela 3 apresenta as frequências observadas (f_o), ou seja, as respostas dos questionários já se considerando a inversão dos valores das respostas das perguntas pares na usabilidade de *design* a partir de SUS, discutido na Seção 3. Visando a imparcialidade, todas as análise utilizarão essas f_o .

Tabela 3. Frequências observadas das usabilidades

| | Usabilidade | | |
|---------------------|-------------|------------|-------|
| | Pedagógica | Design SUS | Total |
| Discordo Totalmente | 1 | 67 | 68 |
| Discordo | 17 | 38 | 55 |
| Neutro | 31 | 35 | 66 |
| Concordo | 61 | 43 | 104 |
| Concordo Totalmente | 130 | 57 | 187 |
| Total | 240 | 240 | 480 |

A pontuação do Boolean Defuse no SUS foi de 90,313, tendo assim sua usabilidade de design classificada com conceito A e adjetivação “Excelente”, ou seja, o resultado aponta que o projeto de design desenvolvido, a partir das heurísticas definidas, está em um nível que permite a distribuição do software no quesito usabilidade de design.

O questionário de avaliação de Boolean Defuse apresentava um campo para recebimento de uma nota de avaliação a partir da experiência de utilização do jogo, a qual será chamada de nota de Experiência Geral. A partir das Situações definidas para avaliar as hipóteses definidas (Seção 3), inicialmente verifica-se a Situação 1, ou seja, se as pontuações atribuídas entre Experiência Geral e SUS obtidas das avaliações entre os grupos diferem estatisticamente. A Tabela 4 apresenta a ANOVA para verificar.

Tabela 4. ANOVA para os valores médios de Experiência Geral e SUS

| RESUMO | | | | | | |
|-------------------------|-----------------|-----------|----------|-----------|----------|-----------|
| Grupo | Contagem | Soma | Média | Variância | | |
| Nota da avaliação geral | 24 | 1950 | 81,25 | 246,196 | | |
| SUS | 24 | 2167,5 | 90,313 | 137,126 | | |
| ANOVA | | | | | | |
| Fonte da variação | SQ | gl | MQ | F | valor-P | F crítico |
| Entre grupos | 985,5469 | 1 | 985,547 | 5,142 | 0,028 | 4,052 |
| Dentro dos grupos | 8816,406 | 46 | 191,661 | - | - | - |
| Total | 9801,953 | 47 | - | - | - | - |

Na comparação entre Experiência Geral e SUS, tem-se que o valor-P $< \alpha$ ($p < 0,05$), logo se confirma (H_{1-S1}), ou seja, as médias obtidas das avaliações entre os grupos diferem estatisticamente. A análise aponta que as avaliações são dependentes da experiência do usuário na utilização de *Boolean Desufe*, com possibilidade de interferência de avaliação pedagógica. Sendo assim, a Tabela 5 analisa-se as avaliações de usabilidade de design e pedagógica.

Tabela 5. ANOVA para os valores médios das usabilidades de design e pedagógica

| RESUMO | | | | | | |
|------------------------|----------|------|--------|-----------|---------|-----------|
| Grupo | Contagem | Soma | Média | Variância | | |
| Usabilidade de Design | 24 | 993 | 41,375 | 52,245 | | |
| Usabilidade Pedagógica | 24 | 1022 | 42,583 | 48,428 | | |
| ANOVA | | | | | | |
| Fonte da variação | SQ | gl | MQ | F | valor-P | F crítico |
| Entre grupos | 17,521 | 1 | 17,521 | 0,348 | 0,558 | 4,051 |
| Dentro dos grupos | 2315,458 | 46 | 50,337 | - | - | - |
| Total | 2332,979 | 47 | - | - | - | - |

Na comparação entre as usabilidades de design e pedagógica, a ANOVA aponta não haver diferença estatística nas médias das avaliações entre os grupos, pois valor-P > α ($p > 0,05$). Portanto, a possibilidade de interferência da usabilidade pedagógica para com a Experiência Geral em relação a SUS, não pode ser verificada, necessitando uma investigação secundária futura para entendimento do por que dessa diferença

Para a análise de inferência estatística da Situação 2, obteve-se ($\chi^2_{calculado}$) = 103,932. Como ($\chi^2_{tabelado}$) = 9,488 para $gl = 4$ e $\alpha = 0,05$, então refuta-se (H_{0-s2}), pois ($\chi^2_{calculado}$) > ($\chi^2_{tabelado}$). Portanto, a usabilidade oferecida por Boolean Defuse fornece contribuição significativa para o processo de ensino e aprendizagem de Expressões Lógicas (H_{1-s2}), tendo em vista a forte dependência ($p < 0,01$) entre as variáveis de usabilidade e o grau de aceitação elevado (Figuras 9 e 10) para as afirmações definidas no questionário de avaliação.

Para verificação da Situação 3, sobre as avaliações do *Boolean Defuse* (usabilidades de *design* e pedagógica, motivacional e experiência do usuário) para cada uma das seguintes variáveis de caracterização de indivíduo: sexo, idade e nível de escolaridade. As variáveis são correspondentes às questões do perfil do usuário. Os dados de usabilidades de *design* e pedagógica são tidos como única base de dados, assim como motivacional e experiência do usuário.

Mas para isso, foi necessário realizar a combinação de categorias adjacentes de modo a aumentar as frequências esperadas nas diversas células, conforme [Siegel 1975, p. 124] e apresentado na Seção 3. Para as usabilidades apenas a variável nível de escolaridade necessitou realizar a combinação, com 5 categorias e 3 níveis de concordância. Para motivação e experiência do usuário, são as combinações: (i) variável sexo com 2 categorias e 3 níveis de concordância; (ii) idade com 2 categorias e 3 níveis de concordância; (iii) para a variável nível de escolaridade as combinações não proporcionaram realizar o teste de dependência.

Os níveis de concordância são relacionados às opções da escala Likert. Os três níveis combinados em categorias são: discordância (combinação de discordo totalmente e discordo), neutro e concordância (combinação de concordo totalmente e concordo).

Na Tabela 6 é apresentado um resumo das análises de dependência das variáveis. A marcação nas células com “Dependente” configura que foi identificada uma dependência, ou seja, no teste Qui-Quadrado o valor calculado de p é inferior a 0,05 ($p < 0,05$).

Tabela 6. Efeito das variáveis de caracterização de indivíduo sobre as avaliações

| Variável (Avaliação - Caracterização) | Usabilidade | |
|---|---------------------|----------------|
| | p (teste χ^2) | Análise |
| Usabilidade (Idade) | 0,254 | Não dependente |
| Usabilidade (Sexo) | 0,569 | Não dependente |
| Usabilidade (Nível de Escolaridade) | 0,657 | Não dependente |
| Motivação e Experiência (Idade) | 0,819 | Não dependente |
| Motivação e Experiência (Sexo) | 0,633 | Não dependente |
| Motivação e Experiência (Nível de Escolaridade) | - | - |

Portanto, chega-se à conclusão que nenhuma das variáveis de caracterização de indivíduo em relação às avaliações de “usabilidades de design e pedagógica” e “motivação e experiência do usuário” tiveram as hipóteses nulas (H_{0-S3}) refutadas, deste modo não existem dependências estatísticas entre o perfil dos usuários e os resultados da avaliação, ou seja, avaliação do jogo educacional Boolean Defuse é independente de variável de caracterização é indivíduo (H_{1-S3}).

Para verificar a Situação 4, se a média das percepções dos usuários quanto ao nível de aprendizagem sobre Expressões Lógicas antes e depois da utilização do jogo *Boolean Defuse* sofreram alguma alteração estatisticamente significativa, tem-se o teste T de Student. Apesar do aumento em todas as médias nas percepções dos objetivos de aprendizagem, é preciso verificar se tal aumento é relevante estatisticamente. Para tal análise têm-se t-crítico = 1,714 ($gl = 23$, $\alpha = 0,05$). Para que seja comprovada uma diferença estatisticamente relevante entre as médias, a estatística t de cada objetivo analisado deve estar fora do intervalo crítico (t_c) -1.714 e 1.714, ou seja, valor-P deve ser menor que 0,05 ($p < 0,05$). Na Tabela 7, têm-se as análises comparativas das médias das percepções de aprendizagem. A marcação nas células com * configura que foi identificada uma diferença, ou seja, o valor calculado de p para o teste t de Student é inferior a 0,05 ($p < 0,05$).

Tabela 7. Avaliação dos objetivos de aprendizagem

| Objetivo de aprendizagem | Momento | Média | Estatística t |
|---|---------|-------|---------------|
| Lembrar: Operadores lógicos | Antes | 4,542 | -0,901 |
| | Depois | 4,667 | |
| Compreender: Operadores lógicos | Antes | 4,417 | -0,720 |
| | Depois | 4,542 | |
| Aplicar: Operadores lógicos | Antes | 3,75 | -1,881* |
| | Depois | 4,083 | |
| Lembrar: Ordem de execução de operações lógicas | Antes | 3,875 | -1,967* |
| | Depois | 4,333 | |
| Compreender: Ordem de execução de operações lógicas | Antes | 4,042 | -1,619 |
| | Depois | 4,417 | |
| Aplicar: Ordem de execução de operações lógicas | Antes | 3,750 | -0,972 |
| | Depois | 4 | |
| Lembrar: Lógica booleana | Antes | 4,375 | -1,282 |
| | Depois | 4,542 | |
| Compreender: Lógica booleana | Antes | 4,250 | -1,904* |
| | Depois | 4,540 | |
| Aplicar: Lógica booleana | Antes | 3,750 | -1,926* |

A análise evidencia que a aprendizagem relacionada a Operadores Lógicos, as médias (μ) percepções Lembrar e Compreender não teve diferença estatística, ou seja, (H_{0-S4}): ($\mu_{antes} - \mu_{depois}$) = 0. Assim como, ordem de execução de operações lógicas

para as percepções Compreender e Aplicar; e Lembrar para Lógica booleana. As demais tiveram diferença nas percepções de aprendizagem ($H_{1-S4}: (\mu_{antes} - \mu_{depois}) > 0, p < 0,05$): percepção Aplicar em Operadores lógicos e Lógica booleana; Lembrar para Ordem de execução de operações lógicas; e Compreender para Lógica booleana.

5. Conclusão

Neste artigo apresentou-se um jogo educacional para o processo de ensino e aprendizagem para Expressões Lógica e discutiu-se sua avaliação, a qual aponta-o como ferramenta que poderá ser utilizada no contexto de cursos de Computação.

A avaliação demonstrou o aumento da percepção de aprendizagem por grande parte dos usuários, em especial, ao aplicar na prática os conhecimentos exigidos no jogo e na compreensão de Lógica Booleana.

Apesar do alto índice SUS atribuído a *Boolean Defuse*, na avaliação, nenhuma das variáveis analisadas obteve 100% de concordância, indicando que o jogo ainda é passível de melhorias. A partir das sugestões, acredita-se que o *software* possa ser melhorado quanto a funcionalidades que visam permitir que usuários menos experientes aproveitem da melhor forma, como, por exemplo: possibilitar a escolha de diferentes níveis de dificuldade; flexibilizar o erro, incorporando o conceito de *life* e vida para situações erradas, mas respeitando a premissa que o enredo trata-se de uma bomba e que operações erradas devem explodí-la, logo essa interação deve transpor o fato de ser uma oportunidade dada pelo NPC *JingSaw*; e possibilitar a mudança de layout para que o usuário possa visualizar apenas um módulo por vez.

Do ponto de vista da análise estatística, aponta-se não refutar H_{0-S4} para a percepção Compreender em Ordem de execução de operações lógicas. No entanto, a estatística está muito próxima do valor de rejeição (6% para 5%), indicando que esta percepção poderia ser, do ponto de vista prático, vista como uma diferença, indicando a necessidade de uma análise futura através de ferramentas de qualidade como, por exemplo, por diagrama espinha de peixe, diagrama de Pareto e histogramas.

Apesar dos resultados positivos apresentados na avaliação, é importante destacar que a pandemia da COVID-19 impossibilitou a realização de um experimento controlado, o qual permitiria maior controle sobre a avaliação, eliminando possíveis ameaças a validade do experimento. Deste modo, não é possível generalizar os resultados obtidos para uma população mais ampla, apesar de os pesquisadores acreditarem que devido a abordagem proposta, os resultados são confiáveis.

Entretanto, ainda que a distribuição do formulário tenha sido dada por convite enviado a e-mails de grupos e mídias sociais, assumindo-se uma amostragem não-probabilística e, por conveniência, na qual é uma técnica de fácil operacionalização e baixo custo, acredita-se que por se tratar de público específico não hajam imposições de diferentes estratos que comprometam o estudo, ou seja, não introduzirá vieses em relação à população.

Referências

- Alves, F. (2015). Gamification - Como criar experiências de aprendizagem engajadoras. Um guia completo. DVS Editora, 2ª ed.
- Bangor, A., Kortum, P. T. and Miller, J. T. (2009). Determining what individual SUS scores mean: adding an adjective rating scale. In: *Journal of Usability Studies*, 4 ed, p.114-123.
- Battistella, P. E., Petri, G., von Wangenheim, C. G., von Wangenheim, A. e Martina, J. E. SORTIA 2.0: Um jogo de ordenação para o ensino de Estrutura de Dados, In: XII Brazilian Symposium on Information Systems, Florianópolis, p. 558-565.
- Berg, A. C. (2007). *Lógica de Programação*. Editora da Ulbra, 3ª ed.
- Brooke, J. (1996). SUS: A “quick and dirty” usability scale. In Jordan, P. W., Thomas, B., Weerdmeester, B. A. and McClelland, A. L. (Eds.), *Usability Evaluation in Industry*. London: Taylor and Francis.
- Cronbach, J. L. (1951). Coefficient alpha and the internal structure of tests. In: *Psychometrika*, v. 16. n. 3, p. 297-334.
- Deterding, S., Sicart, M., Nacke, L., O’Hara, K., and Dixon, D. (2011). Gamification. Using Game-Design Elements in Non-Gaming Contexts. In: *CHI’11 Extended Abstracts on Human Factors in Computing Systems*, p. 2425-2428.
- Deters, J. I., Silva, J. M. C., Miranda, E. M. e Fernandes, A. M. R (2008). O Desafio de Trabalhar com Alunos Repetentes na Disciplina de Algoritmos e Programação. In: *Simpósio Brasileiro de Informática na Educação*.
- Ferreira, B., Rivero, L., LOPES, A., Marquez, A. B. e Conte, T. (2014). UsabiliCity: Um Jogo de Apoio ao Ensino de Propriedades de Usabilidade de Software Através de Analogias. In: *Anais do XXV Simpósio Brasileiro de Informática na Educação*, p. 1273-1282
- Foundation, P. S. (2020). The python language reference, <https://docs.python.org/3/reference/index.htmlthe-python-language-reference>.
- George, D. and Mallery, P. (2003). *SPSS for Windows step by step: A simple guide and reference*. 11.0 update (4th ed.). Boston: Allyn & Bacon. In: Gliem, J. A., Gliem, R. R. (2003). Calculating, Interpreting, and Reporting Cronbach’s Alpha Reliability Coefficient for Likert-Type Scales, *Midwest Research-to-Practice Conference in Adult, Continuing, and Community Education*, The Ohio State University, Columbus, OH, p. 82-88.
- Gomes, A., Joana, H. e Mendes, A. J. (2008). Uma proposta para ajudar alunos com dificuldades na aprendizagem inicial de programação de computadores. In: *Educação, Formação & Tecnologias*, v. 1, n. 1, p. 93-103.
- Gomes, T. e Melo, J. (2016). Jogos Digitais no Ensino de Programação. In: *Anais do Congresso Regional sobre Tecnologias na Educação (Ctrl+E’16)*, CEUR Workshop Proceedings, v. 1667, p. 576–583.

- Johnson, S. (2005). *Surpreendente!: A televisão e o videogame nos tornam mais inteligentes*. Rio de Janeiro: Elsevier.
- Júnior, A. F. M., Vieira, F. M. S. e Lacerda, M. M. (2017). Reflexões Sobre Avaliação de Games Educacionais. In: XII Seminário Jogos Eletrônicos – Educação – Comunicação (SJEEC'17), p. 11–20.
- Krasner, G. E. and Pope, S. T (1988). A description of the model-view-controller user interface paradigm in the smalltalk80 system. In: *Journal of Object-oriented Programming - JOOP*, 1.
- Levin, J. e Fox, J. A. (2004). “Estatística para Ciências Humanas”, São Paulo: Prentice Hall, 9 ed.
- Likert, R. (1932). A technique for the measurement of attitudes. In: *Archives of Psychology*. n. 140, p. 44-53.
- Liu, Y., Alexandrova, T., and Nakajima, T. (2011). Gamifying intelligent environments. In: *Proceedings of the 2011 international ACM workshop on Ubiquitous meta user interfaces*.
- Massário, M. S., Barreto, C. H. C., Knoll, G. F. e Ghisleni, T. (2019). Gamificação como prática de ensino. In: *Research, Society and Development*, v. 8, n. 7.
- Navarro, G. (2013). Gamificação: A transformação do conceito do termo jogo no contexto da pós-modernidade. Trabalho de Conclusão de Curso (Especialização em Mídia, Informação e Cultura) - CELACC/ECA – USP, 26 p.
- Neto, S. R. S., Silva, E. G., Barbosa, A. F., Santos, W. O. Souza, A. A., Oliveira, M. L. S. e Balieiro, A. M. (2015). Avaliação do Jogo Educativo Mundo de Euclides: Uma abordagem multi-perspectiva. In: *Anais dos Workshops do IV Congresso Brasileiro de Informática na Educação*, p. 1114–1123.
- Nielsen, J. (1994). “10 Usability Heuristics for User Interface Design”, <https://www.nngroup.com/articles/ten-usability-heuristics/>, Dezembro.
- Nogueira, J. E. S. S., Veras, W. T. A., Silva, J. E. N. & D’Emery, R. A. (2020). “Boolean Defuse: Avaliação de usabilidade de um jogodesktop para apoio no aprendizado de expressões lógicas”. In: *XXVIII Ciclo de Palestras sobre Novas Tecnologias na Educação*, Porto Alegre, Setembro, p. 40-49.
- Nogueira, J. E. S. S. e D’Emery, R. A. (2021). Avaliação de Usabilidade do Jogo Educacional Boolean Defuse v. 2.0. Relatório Técnico – Universidade Federal Rural de Pernambuco, Serra Talhada, Fevereiro.
- Novak, J. (2012). *Game Development Essentials*. Delmar Cengage Learning, 3ª ed..
- Oliveira, H. e Savoine, M. (2011) “Aplicação do Método de Heurística no Sistema Colaborativo HEDS”, In: *Revista Científica do ITPAC*, v.4, n.3, P11, Araguaína - Brasil.
- Oracle (2009). *Java Programming Language*, <https://docs.oracle.com/javase/8/>.
- Plackett, R. L. (1983). Karl Pearson and the Chi-Squared Test. In: *International Statistical Review*. v. 51, n. 1, p. 59–72.

- Prensky, M. (2001). Digital Natives Digital Immigrants. In: PRENSKY, Marc. On the Horizon. NCB University Press, v. 9, n. 5, October.
- Reategui, E. e Finco, M. D. (2010). Proposta de Diretrizes para Avaliação de Objetos de Aprendizagem Considerando Aspectos Pedagógicos e Técnicos. In: Renote – Revista Novas Tecnologias na Educação, v. 8, n. 3.
- Revythi, A. and Tselios, N. (2017). Extension of technology acceptance model by using system usability scale to assess behavioral intention to use e-learning. In: Education and Information Technologies, v. 24, p. 2341-2355.
- Savi, R., von Wangenheim, C. G. and Borgatto, A. F. (2011). A model for the evaluation of educational games for teaching software engineering. In: 25th Brazilian Symposium on Software Engineering (SBES 2011), Sao Paulo, Brazil, September, IEEE Computer Society p. 194–203.
- Siegel, S. (1975). Estatística Não-paramétrica para as Ciências do Comportamento. São Paulo: McGraw-Hill.
- Silva, M. A. A. e Dantas, A. (2014). KLouro: Um jogo educacional para motivar alunos iniciantes em programação. In: Anais do XXV Simpósio Brasileiro de Informática na Educação (SBIE 2014), p. 702-711.
- Silva, S. S., Calazans, A. T. S. e Silva, B. M. (2020). Construção e avaliação de um jogo educacional digital de modelagem de dados para o ambiente universitário. In: XXXVI Congresso da Sociedade Brasileira de Computação, p. 2036-2045.
- Student (1908). The probable error of a mean. In: Biometrika.
- Vahldick, A. e Mattos, M. (2008). Relato de uma experiência no ensino de algoritmos e programação utilizando um framework lúdico. In: Anais do XIX SBIE-Simpósio Brasileiro de Informática na Educação.
- Vahldick, A. e Pereira, G. N. Y. (2019). Conduzindo os alunos a resolverem exercícios de programação através da verificação de seus objetivos durante a execução do programa. In: Anais do XXX Simpósio Brasileiro de Informática na Educação, p. 71-80.
- Vasconcelos, J. (2017). Avaliação heurística do moodle, http://www.jvasconcellos.com.br/unijorge/wp-content/uploads/2011/11/questionario_avaliacao_heuristica_final.doc/.
- Vieira, S. (2006). “Análise de variância: ANOVA”. São Paulo: Atlas.
- WHO. (2021). “WHO Coronavirus Disease (COVID-19) Dashboard”, World Health Organization, <https://covid19.who.int/>, Fevereiro.